VU Research Portal

Identification on a manifold of systems

Peeters, R.L.M.

1992

Faculteit der Economische Wetenschappen en Econometrie

Serie Research Memoranda

Identification on a Manifold of Systems

R.L.M. Peeters

Research Memorandum 1992-7
maart 1992

vrije Universiteit *amsterdam*

Identification on a Manifold of Systems *

Ralf Peeters †
Free University, Amsterdam

Summary of presentation at 11th Benelux Meeting, Veldhoven, The Netherlands

March 4–6, 1992

Abstract

When considering the space of linear multivariable systems of fixed finite order n under i/o–equivalence, it is a known fact (cf. Hazewinkel [21]) that we are dealing with a differentiable manifold. This manifold can be covered with a finite number of overlapping parameter charts and there exist methods to select an appropriate chart on–line (cf. Van Overbeek and Ljung [34,35]). Moreover, it is possible to endow this manifold with various Riemannian metrics, expressing the notion of distance between systems in a coordinate free way (cf. Hanzon [18,19]). If one adopts a prediction error criterion to measure the quality of a model describing a given set of data, then, in the case of batch (off–line) identification, the problem of system identification boils down to the deterministic problem of minimizing a nonlinear least squares criterion over a Riemannian manifold.

There exist several methods to minimize a nonlinear least squares criterion over Euclidean space, of which the methods of Gauss–Newton and of Levenberg–Marquardt are the most important as they exploit the structure of the problem. In the manifold case, however, there has not been paid much attention to this problem yet, leaving us with only general approaches towards function minimization.

In this paper we discuss a Riemannian interpretation of the Gauss–Newton algorithm and we describe a Riemannian version of the Levenberg–Marquardt algorithm. We next formulate the identification problem mentioned above and indicate how to obtain Riemannian metrics and overlapping parametrizations. In the last section we describe a simulation experiment and we present and discuss the results. It is observed that there exist Riemannian gradient methods which can exhibit superlinear convergence properties. In particular this holds true if one uses the Fisher information matrix to define a Riemannian metric.

1 Nonlinear Least Squares

1.1 Nonlinear Least Squares on Euclidean n–space

We are interested in *minimizing* (locally) the following criterion function $\Phi : \mathbf{R}^n \to \mathbf{R}$ defined by

$$\Phi(x) = \frac{1}{2}\|f(x)\|^2 = \frac{1}{2}\sum_{i=1}^{m}[f^i(x)]^2$$

where $f : \mathbf{R}^n \to \mathbf{R}^m$ denotes the *residual mapping* with corresponding coordinate functions (the so–called *residuals*) $f^i : \mathbf{R}^n \to \mathbf{R}$, $(i = 1, \ldots, m)$, and $\|\cdot\|$ denotes the Euclidean norm (on $\mathbf{R}^m$). We assume f to be at least *twice* continuously differentiable, in order to be able to apply Newton's method for minimization and to compare with it.

The associated Jacobian mapping is denoted by $J : \mathbf{R}^n \to \mathbf{R}^{m\times n}$ and defined in each point $x \in \mathbf{R}^n$ by:

*This research was carried out as part of NWO research project 611-304-019.

†Address: Free University, Department of Economics and Econometrics, De Boelelaan 1105, 1081 HV Amsterdam, The Netherlands. E-mail: ralf@sara.nl.

$$J(x) = \begin{pmatrix} \frac{\partial f^1}{\partial x^1}(x) & \cdots & \frac{\partial f^1}{\partial x^n}(x) \\ \vdots & & \vdots \\ \frac{\partial f^m}{\partial x^1}(x) & \cdots & \frac{\partial f^m}{\partial x^n}(x) \end{pmatrix}$$

We adopt the convention that coordinates (and coordinate functions) are indexed by a superscript, whereas quantities changing every iteration are indexed by a subscript k, denoting the iteration number.
A second assumption we make is that Φ possesses only *isolated* (local) minima, and a third that there *exists* at least one local minimum at x_*, say.

The above problem is referred to as the *(nonlinear) least squares problem.* An important subclass consists of those situations where the residual mapping is *linear* in x. These are the *linear least squares problems.* Though our interest is in the nonlinear case, we shall be paying attention to the linear case as well because of the fact that most methods available for (iteratively) solving the nonlinear least squares problem proceed by solving a sequence of linear least squares problems.
The linear least squares problem can be solved analytically (and numerically efficient), whereas the nonlinear least squares problem in general cannot.

There exists a large number of minimization methods that are designed to handle the problem of minimizing an arbitrary function over $\mathbf{R}^n$. Well known methods are *gradient methods (steepest descent), Newton and quasi-Newton methods, conjugate gradient methods, trust-region methods and direct search methods* (that make use of function values only and not of derivatives). See e.g. Dennis and Schnabel [7], Bard [4]. A disadvantage for applying such general algorithms in this particular case of nonlinear least squares is that none of them exploits the *structure* of the problem, i.e. the fact that we are dealing with a *sum of squares.*
There exist, however, also methods that do exploit this structure. They are designed especially for the nonlinear least squares problem, as it happens that this problem occurs quite often in practice. The most well-known such method is the *method of Gauss-Newton*, of which there exist several variants (concerning the choice of an appropriate step-size), whereas an alternative (or rather extension) is provided by the *method of Levenberg-Marquardt.* Both exploit the available structure and local information in each iteration in a relatively efficient way. They make use of *first order information* only as opposed to Newton's method which requires knowledge of the Hessian (that is, *second order information*) at each iterate. We shall restrict the discussion to the methods of Gauss-Newton and of Levenberg-Marquardt and point out some relationships with Newton's method. An excellent treatment of the subject of least squares can be found in Chapter 10 of Dennis and Schnabel [7].

The *gradient* of Φ at point x is denoted by $g(x)$ and satisfies

$$g(x) = J(x)^T f(x)$$

where superscript ^{T} denotes transposition. According to our definition, $g(x)$ is a *column* vector of dimension n.
The *Hessian* of Φ at point x is denoted by $H(x)$ and given by

$$H(x) = J(x)^T J(x) + \sum_{i=1}^{m} f^i(x) \frac{\partial^2 f^i}{\partial x^2}(x)$$

Here $\frac{\partial^2 f^i}{\partial x^2}(x)$ denotes the Hessian of the ith coordinate function, that is

$$\frac{\partial^2 f^i}{\partial x^2}(x) = \begin{pmatrix} \frac{\partial^2 f^i}{\partial x^1 \partial x^1}(x) & \cdots & \frac{\partial^2 f^i}{\partial x^1 \partial x^n}(x) \\ \vdots & & \vdots \\ \frac{\partial^2 f^i}{\partial x^1 \partial x^n}(x) & \cdots & \frac{\partial^2 f^i}{\partial x^n \partial x^n}(x) \end{pmatrix}$$

When approximating $\Phi(x)$ by its second order Taylor series expansion one obtains a quadratic approximating function which can be minimized exactly, analytically. The (standard) Newton method

exploits this fact and proceeds by taking the next estimate x_{k+1} of the local optimum x_*, starting from a current estimate x_k, as the point that minimizes this approximating quadratic function:

$$x_{k+1} = x_k - H(x_k)^{-1} g(x_k)$$

There are a few aspects about this method that need further attention.
(1) The negative gradient direction is always a *decreasing* direction, that is, if we take a positive step in this direction of sufficiently small size, we can always obtain a decrease of function value. This is *not* necessarily true for the Newton direction $-H(x_k)^{-1}g(x_k)$. A sufficient condition would be positive definiteness of $H(x_k)$, which is not guaranteed. On the other hand, at a local (isolated) minimum one will always have that $H(x_*)$ is at least positive semi–definite, so that good *local* convergence properties can be expected.
(2) The method is not well–defined for singular $H(x_k)$. In such a case an alternative strategy becomes necessary, that is, one must specify what to do in such cases.
(3) Even if the Newton direction is decreasing, the step taken above may be too large. An extra step–size controlling parameter is then needed. The common procedure is to perform a line minimization in the proposed direction to arrive at a better estimate for x_*. Accordingly, the recursion takes the form

$$x_{k+1} = x_k - \alpha_k H(x_k)^{-1} g(x_k)$$

with $\alpha_k > 0$ denoting the step–size controlling parameter. Such a method is referred to as a *damped* Newton method.
(4) The method is *expensive* in the sense that it requires second order information.

The method of Gauss–Newton yields an improvement of Newton's method with respect to points (1) and (4). Here we define the *Gauss–Newton matrix* $G(x)$ as

$$G(x) = J(x)^T J(x)$$

The associated iterative scheme becomes

$$x_{k+1} = x_k - G(x_k)^{-1} g(x_k)$$

Notice that: (1) The search direction is now always a decreasing direction since $G(x)$ is, by construction, positive semi–definite. (2) The method is cheaper, since only *first order* information is required.
However, local convergence properties near an optimum x_* for which $H(x_*)$ is positive definite may be *worse* than for Newton's method, depending on the accuracy by which $G(x_*)$ approximates $H(x_*)$. There are two situations where this is for sure *not* the case, namely (a) the linear case, (b) the case where the optimum criterion value is (close to) zero. It is easily seen that in those situations the term neglected by $G(x_*)$, as an approximation of $H(x_*)$, is (almost) zero. But objections (2) and (3) against the (standard) Newton method still apply to this (standard) Gauss–Newton method also.
There are several alternative ways of arriving at the Gauss–Newton method, which provide more insight in its properties. Above we took the standpoint of regarding $G(x)$ as a (positive semi–definite) *approximation to the Hessian* $H(x)$. A related, but slightly different point of view is obtained by considering an alternative local approximation of $\Phi(x)$, instead of via a second order Taylor series expansion. We can apply a *quasi–linearization* strategy, by which we *linearize* the *residuals* f^i with respect to x. This leads to the approximating *linear* least squares criterion (at x_k):

$$\Phi_k(x) = \frac{1}{2}\|f(x_k) + J(x_k)(x - x_k)\|^2$$

Then x_{k+1} as defined above by the Gauss–Newton scheme is easily seen to minimize $\Phi_k(x)$.
This point of view provides us with a meaningful way to proceed in cases where $G(x_k)$ is singular, or equivalently, where $J(x_k)$ does not have full column rank n. In such a situation we can try to solve the approximating linear least squares problem anyway. Though then there will not be a unique solution for the original problem anymore, we can again obtain a unique answer if we impose the

additional requirement of finding the minimizing step of minimal length. Then a solution against objection (2) with respect to Newton's method has been obtained. However, objection (3) remains valid and the presence of a step–size controlling parameter might still prove to be essential.
There exists yet another point of view towards the nonlinear least squares problem which also suggests in a natural way the applicability of the Gauss–Newton method. Here, however, also the need for a step–size controlling parameter becomes evident and does not appear as an "artifical" device to prevent against undesirable situations. We discuss it in the next subsection, as it involves the use of Riemannian manifolds.

The *Levenberg–Marquardt method* can be viewed as an extension of the Gauss–Newton method in the following sense. In the classical approach towards standard Gauss–Newton (i.e. without a step–size controlling parameter) it can happen in certain situations that the series of criterion values obtained is non–decreasing. This corresponds to the fact that the steps that are taken are *too large*, since the search directions are guaranteed decreasing. Therefore, one can try to *balance* the criterion function with the second objective of keeping the step–size small. We therefore obtain a local criterion of the form

$$\tilde{\Phi}_k(x) = \frac{1}{2}(\|f(x_k) + J(x_k)(x - x_k)\|^2 + \lambda_k \|D_k(x - x_k)\|^2)$$

where λ_k is the (non–negative) *Levenberg–Marquardt parameter* that acts as a *balancing parameter* between the two criteria, and D_k denotes a non–singular *weighting* (or *scaling*) matrix, so that $\|D_k p\|$ represents the relevant *norm* (or length) of a step p.
Again, this leaves us with a *linear* least squares problem, but this time with the nice property of having a *unique* solution always, provided D_k is indeed non–singular. This is seen by rewriting $\tilde{\Phi}_k(x)$ as

$$\tilde{\Phi}_k(x) = \frac{1}{2}\left\| \begin{pmatrix} f(x_k) \\ 0 \end{pmatrix} + \begin{pmatrix} J(x_k) \\ \sqrt{\lambda_k} D_k \end{pmatrix} (x - x_k) \right\|^2$$

The solution to this problem is now given by

$$x_{k+1} = x_k - (J(x_k)^T J(x_k) + \lambda_k D_k^T D_k)^{-1} J(x_k)^T f(x_k)$$

Of course, we have created additional problems, namely how to choose λ_k and D_k. There are again several approaches possible to this, but the most elegant and at the same time very powerful one is provided by Moré [33]. He followed a *trust–region* approach towards the nonlinear least squares problem and came up with a very robust implementation of the Levenberg–Marquardt algorithm. The basic ideas are the following.
Together with an estimate x_k for x_* we specify a trust–region of the form

$$\{p \in \mathbf{R}^n \mid \|D_k p\| \leq \Delta_k\}$$

that is, we specify a non–singular $n \times n$ matrix D_k and a positive scalar Δ_k. This trust–region indicates the area where the proposed step as generated via quasi–linearization, so by Gauss–Newton, is required to lie in order to be acceptable. (For this region we "trust" the quasi–linearization approach.) Accordingly, if an unrestricted Gauss–Newton step would lead us outside the trust–region, we perform *constrained minimization*, namely, we minimize the approximating linear least squares criterion over the (quadratic) trust–region. This then leads to a problem that can be solved iteratively, again via a sequence of linear least squares problems, in a highly efficient way, provided we treat the trust–region *size* Δ_k flexibly. (Notice that also in case the Gauss–Newton step is acceptable we can view the optimization problem as constrained; only then the solution is much easier to obtain.) Together with the calculation of a new step comes the calculation of a new trust–region size. Moré has developed a procedure for adapting this size via inspection of the actually achieved improvement in the criterion value as compared to the predicted improvement (via the linearization). The choice of D_k however remains relatively *heuristic*. Moré proposes various scaling strategies, all corresponding to diagonal matrices, but none of them is based in a mathematically solid way. Notice that if one chooses $D_k = I$, irrespective of the estimate x_k and irrespective of

the iteration number, then we are in the situation of *non-scaling*, which was standard before Moré. In Figure 1 we present a flow-diagram of the Levenberg-Marquardt algorithm as implemented by Moré.
We remark that an alternative interpretation of the Levenberg-Marquardt method is obtained by viewing each generated search direction as an *interpolation* between the method of Gauss-Newton and the method of steepest descent (in case $D_k = I$). According to Marquardt [32] it is observed for a variety of problems that the angle, γ, between the steepest descent direction and the Gauss-Newton direction usually falls in the range $80° < \gamma < 90°$. This provides another rationale for using the Levenberg-Marquardt method.

1.2 Nonlinear Least Squares on a Riemannian n-Manifold

We now address a slight extension of the nonlinear least squares problem discussed in the previous subsection, where the domain of the criterion function is no longer required to be $\mathbf{R}^n$, but assumes the more general structure of a Riemannian n-manifold. Thus we are dealing with the problem of minimizing a function $\Phi : M \to \mathbf{R}$, where M is an n-dimensional Riemannian manifold, defined by

$$\Phi(x) = \frac{1}{2}\|f(x)\|^2 = \frac{1}{2}\sum_{i=1}^{m}[f^i(x)]^2$$

where $f : M \to \mathbf{R}^m$ still denotes the *residual mapping* with corresponding coordinate functions $f^i : M \to \mathbf{R}$, $(i = 1,\ldots,m)$, and $\|\cdot\|$ denotes the Euclidean norm (on $\mathbf{R}^m$). As before we assume f to be at least twice continuously differentiable.
One can consider Euclidean n-space (i.e. $\mathbf{R}^n$ with the Euclidean metric) to be a Riemannian n-manifold, so that the previous case is contained in the present one.
Again, there are several general techniques available for the minimization of a function over a (Riemannian) manifold. Techniques exploiting the Riemannian metric have been proposed only relatively recent (see e.g. Luenberger [31], Lichnewsky [27,28], Gabay [11]), where the basic motivation for this kind of research stems from the field of *constrained* optimization, in which the constraints are supposed to constitute a (Riemannian) differentiable manifold.
In practice, the approach commonly followed seems to be to use overlapping coordinate charts that cover the manifold, which are treated each in a "Euclidean" way by applying standard optimization techniques. (Often one resorts to selecting just one chart, already *before* the actual minimization is carried out.) Such an approach has the advantage of being applicable also if there is *no* a priori Riemannian metric available. On the other hand one can argue that the approach is rather unelegant because of the following. (1) It is a basic result from differential geometry that each (smooth) differentiable manifold can be endowed with a Riemannian metric. Therefore, if there is no Riemannian metric available yet, one can in principle define one. (2) The Euclidean metrics defined on the coordinate charts usually are not matching to one another in the areas where the charts overlap. The local metric involved is therefore highly dependent on the choice of coordinates. The same applies to the expected convergence properties of optimization algorithms. Notice that application of the *same* optimization method, starting from the *same* point on the manifold but with *another* coordinate chart will generally lead to *different* iteration paths on the manifold. An illustration of this observation is provided by Figure 2, for the example of minimization of a quadratic criterion by the method of steepest descent. In case the coordinates are chosen appropriately the optimum can be reached in one step. Otherwise an iteration path will be followed, showing that the iteration paths on the manifold are dependent on the choice of coordinates.
In Lichnewsky [27] one can find certain generalizations of Newton's method and conjugate gradient methods to the manifold case. The idea of Riemannian steepest descent is apparently also well-known, although practical application of this method seems to be rather limited due to the bad convergence reputation of steepest descent in the Euclidean case. However, it is unclear whether there have been designed algorithms that act on a manifold *in a coordinate independent way* for the case of *nonlinear least squares* especially. It turns out that the method of *Gauss-Newton* can be interpreted as (largely) coordinate free and that the method of *Levenberg-Marquardt* can be modified to a "Riemannian" version as well. This will be the subject of the rest of this section.

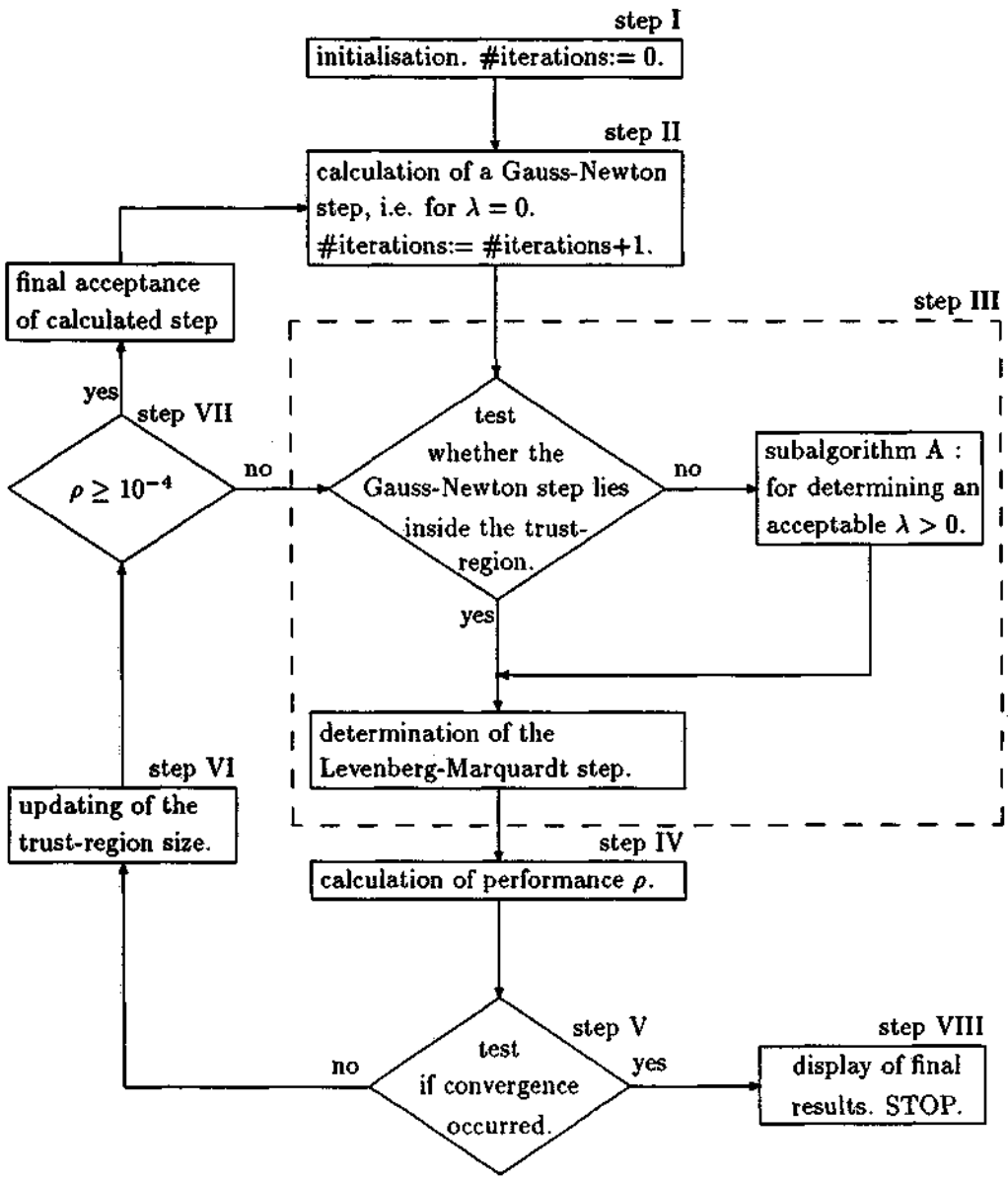


Figure 1. Flowdiagram of the Levenberg–Marquardt algorithm, according to Moré's trust-region implementation.

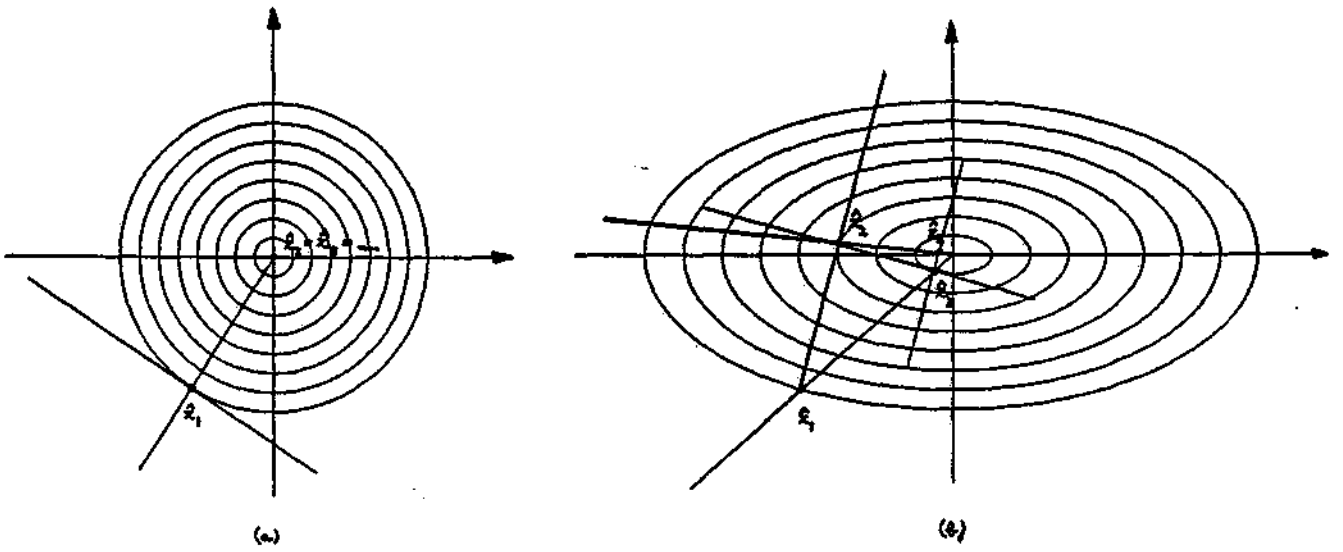


Figure 2. Scaling dependence for the iteration paths of the method of steepest descent — (a) circular contour plots, (b) elliptic contour plots.

We now give an interesting interpretation of the Gauss–Newton method, described in the previous subsection, which shows that we can look upon its behaviour as *largely coordinate independent*, (i.e. virtually regardless of the actual coordinate charts being used). To make this more precise we have to look at the *residual mapping* more closely and interpret the original minimization problem as the problem of finding the point in the *image* of f that is closest to $0 \in \mathbf{R}^m$ (at least locally). Regarding an appropriate open neighbourhood of a local isolated minimum x_* at which the Jacobian has full column rank n, we generally will find that the image of this neighbourhood forms a *differentiable manifold* of dimension n, embedded in $\mathbf{R}^m$. This differentiable manifold *naturally* admits a Riemannian metric, by taking the infimum path length metric induced by the Euclidean metric on the image space $\mathbf{R}^m$. Of course, since the residual mapping is locally one–to–one, this Riemannian metric can be transferred to the domain space: we obtain a Riemannian metric on the open neighbourhood of x_*, regarded as a differentiable manifold by identification of the domain and image. This concept is illustrated by Figure 3. It shows that local coordinates for M can be used as local coordinates for $f(M)$ also. Though the choice of coordinates may be varying, the image $f(M)$ of M in $\mathbf{R}^m$ is fixed.

On a Riemannian manifold the appropriate generalization of the concept of gradient is the *Riemannian gradient*, obtained as the *maximizing normalized tangent direction* with respect to the criterion function, where the normalization is in terms of the Riemannian metric. (Cf. Abraham and Marsden [1].) It turns out that the *Riemannian steepest descent direction* (the negative Riemannian gradient) in the present case coincides exactly with the *Gauss–Newton direction*. And as we are dealing essentially with a *gradient* method in this point of view, the use of a step–size controlling parameter is natural. The choice of this parameter could then be based, as usual, on a "line" minimization procedure in the obtained direction, which in the manifold case should be replaced by a *geodesic search*. For a more detailed account of these statements we refer to Hanzon and Peeters [20]. We remark the following important features of this point of view.

(1) The Gauss–Newton method is *not* obtained via approximation of the original criterion function

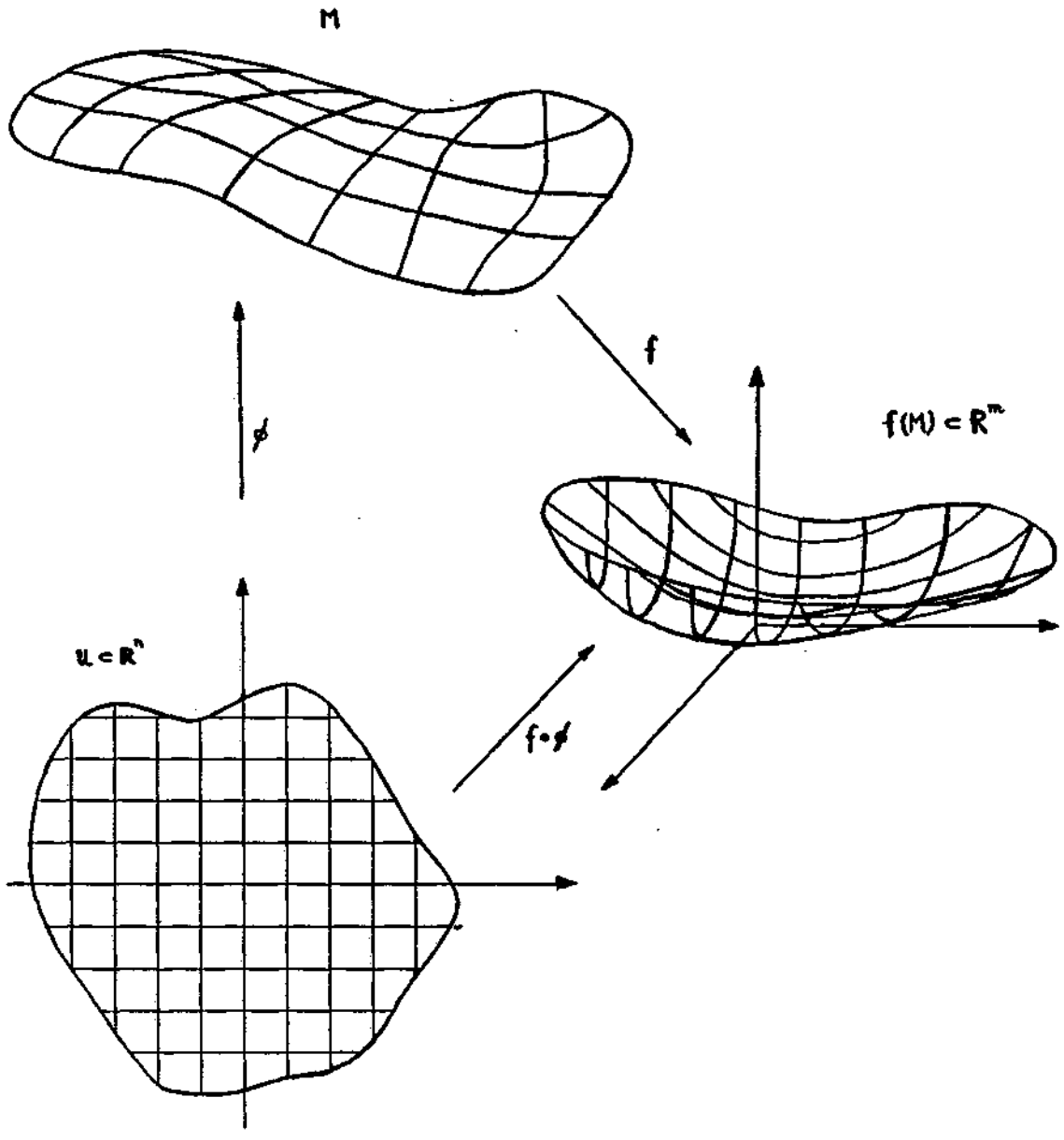


Figure 3. Relation between M, $f(M)$ and local coordinate neighbourhood (U, ϕ).

$\Phi(x)$ of any kind, but as an *exact* (Riemannian) gradient method. In practice the only approximation that will be made is on the level of the calculation of a step, for which a geodesic search should be performed, so that accordingly we should follow a (generally nonlinear) curve in the original domain space. This will usually be replaced by a linear approximation, yielding the *damped Gauss–Newton* method as we know it.
(2) The Riemannian metric induced by the Euclidean metric in the image space is a conceptually appealing one, since it relates to our intuition that two points x_1 and x_2 should be considered as "far apart" (in the present problem setting) if their corresponding residual vectors $f(x_1)$ and $f(x_2)$ differ "a lot". Thus, this (locally) Riemannian metric is quite naturally related to the nonlinear least squares problem.
(3) Notice that the induced Riemannian metric is defined completely independent of a possibly available Riemannian metric on the domain. This not only relates to the independence of the method of the coordinate charts, but also broadens its range of applicability to the case of differentiable manifolds without an a priori Riemannian metric.

The next point we want to make clear is that the above point of view with respect to the method of Gauss–Newton helps us to construct a *Riemannian version of the Levenberg–Marquardt method.* Indeed, observing that the Gauss–Newton method is (largely) coordinate independent, combined with the earlier interpretation of the Levenberg–Marquardt method as an interpolation between Gauss–Newton and steepest descent, suggests the idea of replacing the coordinate dependent gradient by its coordinate free counterpart: the Riemannian gradient. The Riemannian Levenberg–Marquardt algorithm can accordingly be viewed as an interpolation between the Gauss–Newton and the Riemannian steepest descent method.
However, also Moré's approach now can be given a more solid basis. On a Riemannian manifold all search directions are associated with *tangent vectors.* The trust–region idea then can be applied to the *tangent space* to the manifold at x_k instead of directly to the space of coordinates. In this approach a trust–region consists of a collection of tangent vectors that can be accepted as search steps, when using *normal coordinates.* Thus, the *length* of a tangent vector is measured in terms of the *Riemannian metric* on the manifold. This Riemannian metric expresses the local intuition about distance and provides a natural choice of "scaling". More precisely stated, it is possible to define a trust–region on the *tangent space* to the manifold at each point. This can be translated to a *domain on the manifold* by introducing *normal coordinates.* Each tangent vector (with admissable Riemannian length) then corresponds to a step on the manifold by following a geodesic in the appropriate direction with suitable length.
Therefore, we obtain a natural choice for the matrix D_k in Moré's algorithm: it should be chosen such that

$$D_k^T D_k = R(x_k)$$

where $R(x)$ denotes the matrix associated with the Riemannian metric tensor at point x of the manifold, in terms of the local coordinates being used. Notice that although there is freedom left in the precise choice of D_k, the trust–region is well–defined (c.q. uniquely), since the trust–region consists of all tangent vectors satisfying

$$\|D_k p\| \leq \Delta_k$$

which is equivalent to

$$p^T D_k^T D_k p \leq \Delta_k^2$$

The freedom left in D_k can be exploited to achieve good numerical behaviour. Since $R(x)$ is always positive definite by definition (it represents a Riemannian metric) we can for instance choose D_k as a Choleski factor of $R(x_k)$, but also LD–factorization or singular value decomposition might be of use.
The standard Levenberg–Marquardt approach (without scaling) can now be interpreted as corresponding to the situation where we accept the Euclidean metric everywhere. Depending on the problem at hand, we should try to choose a Riemannian metric on M such as to express our intuition about the distance between points in the domain. For a more detailed description of this Riemannian version of the Levenberg–Marquardt method we refer to Peeters [36].

2 Off–Line Identification on a Manifold of Systems

2.1 Manifolds of Systems

In this subsection we state a number of facts about the manifold structure of certain classes of linear systems. We restrict ourselves to the case of *linear, stochastic, time–invariant, asymptotically stable, minimum phase, multivariable, discrete–time systems of fixed, finite order n, driven by Gaussian white noise.* These correspond, in *state–space representation*, to recursive systems of equations of the form (the so–called *innovations form*)

$$\begin{cases} x(t+1) = Ax(t) + B\epsilon(t) \\ y(t) = Cx(t) + \epsilon(t) \end{cases} \qquad (t \in \mathbf{Z})$$

with $A \in \mathbf{R}^{n\times n}$, $B \in \mathbf{R}^{n\times p}$, $C \in \mathbf{R}^{p\times n}$, $x(t) \in \mathbf{R}^n$, where the eigenvalues of A and $A - BC$ are required to lie inside the open unit disk $\{s \in \mathbf{C} \,|\, |s| < 1\}$, where matrix triple (A, B, C) is required to be *minimal* (i.e. the system representation is both *observable* and *reachable*) and where $\epsilon(t)$ is required to be *stationary Gaussian white noise*: $E\epsilon(t) = 0$, $E\epsilon(t)\epsilon(s)^T = \delta_{ts}\Sigma_\epsilon$, with $\Sigma_\epsilon > 0$. We assume both the (stochastic) inputs and outputs to be p–dimensional.
The motivation for these assumptions is based on the *Wold Decomposition and Representation Theorem*, see e.g. Hannan and Deistler [15] from which the following is taken:

Theorem Every stationary process $y(t)$ can be represented in a unique way as

$$y(t) = u(t) + v(t)$$

where $u(t)$ and $v(t)$ are obtainable as linear transformations of $y(t)$, where $H_u(t) \subset H_y(t)$, $H_v(t) \subset H_y(t)$, where $Eu(t)v(s)^T = 0$ and where $u(t)$ is linearly regular and $v(t)$ is linearly singular. Furthermore, every linear regular process $u(t)$ can be represented as

$$u(t) = \sum_{j=0}^{\infty} K(j)\epsilon(t-j), \qquad \sum_{j=0}^{\infty} \|K(j)\|^2 < \infty$$

where $H_u(t) = H_\epsilon(t)$ and where $\epsilon(t)$ is white noise. Thus $\epsilon(t)$ are linear innovations of $y(t)$.

In this theorem "stationary" refers to *wide sense stationarity*. Further, $H_y(t)$ refers to the Hilbert space in $L^2(\Omega, \mathcal{F}, P)$ (the Hilbert space of square–integrable complex random variables over the underlying probability space $(\Omega, \mathcal{F}, P)$) spanned by the components $\{y_i(t-j) \,|\, i = 1, \ldots, p; j \geq 0\}$. The Hilbert spaces $H_u(t)$, $H_v(t)$ and $H_\epsilon(t)$ are defined analogously. A stationary process $u(t)$ is called *linearly regular* if $Eu(t) = 0$ and if for the best linear predictor $u(t+\tau|t)$ of $u(t+\tau)$ (for $\tau > 0$) based on $u(s)$, $s \leq t$, the relation

$$\lim_{\tau\to\infty} u(t+\tau|t) = 0$$

holds, or equivalently if

$$S = \bigcap_{t\in\mathbf{Z}} H_u(t) = \{0\}$$

A stationary process $v(t)$ is called *linearly singular* if

$$v(t+\tau|t) = v(t+\tau) \quad \text{a.s.}$$

holds, or equivalently if

$$S = \bigcap_{t\in\mathbf{Z}} H_v(t) = H_v$$

where H_v is the time domain of $v(t)$, that is, the Hilbert space spanned by $\{v_i(t) \,|\, i = 1, \ldots, p; t \in \mathbf{Z}\}$.
A proof of this theorem can be found in Hannan [14] or Rozanov [39].
It is worth–while to notice that we can choose in particular $\epsilon(t)$ as $u(t) - u(t|t-1)$.

We then restrict ourselves to *linearly regular* processes $y(t)$, on the argument (cf. Hannan and Deistler [15]) that if only a part of one realization of $y(t)$ is available, then the linearly singular part could be considered deterministic and could be removed, so that $u(t)$ in the theorem above is identical to $y(t)$ and $v(t) = 0$. Then via the Wold representation one can construct an *infinite dimensional* state-space system of the form given above, yielding an alternative representation of the process $y(t)$. The *spectral density* of a linear regular process $y(t)$ always exists, and is given by

$$f_y(\omega) = \frac{1}{2\pi} k(e^{i\omega}) \Sigma_\epsilon k^*(e^{i\omega})$$

where

$$k(z) = \sum_{j=0}^{\infty} K(j) z^j$$

denotes the transfer function of the system and $\Sigma_\epsilon = E\epsilon(t)\epsilon(t)^T$ is the covariance matrix of $\epsilon(t)$. It is *no restriction of generality* to assume, as we do, that

$$k(0) = I$$

It however *is* a restriction of generality to assume that $\Sigma_\epsilon > 0$, since in the general case only positive *semi*-definiteness is ensured. Such an assumption is equivalent to the assumption that $y(t)$ is a *full rank process*, meaning that the spectral density f_y has full rank p almost everywhere (a.e.). We nevertheless shall make such an assumption for convenience, noticing that such situations are quite unlikely to occur in practice (in accordance with Hannan and Deistler).
In case f_y is *rational*, we also will have a *rational* transfer function $k(z)$, i.e. of *finite* McMillan degree, say n. This McMillan degree will then correspond to the minimal dimension of the state $x(t)$ that is required to represent the system and this is called the system *order*.
In such a case we then can *always* find an asymptotically stable matrix A and matrices B and C such that $A - BC$ is stable. Therefore, only the assumption about *asymptotic* stability of $A - BC$ is again a restriction of generality. Again this excludes only a "thin" set, and the advantages of the assumptions will be apparent from the sequel (in particular, our predictor error filters will be asymptotically stable).

Returning to our set of *minimal* state-space systems of order n we can associate with each system in this set its transfer function $k(z)$ or rather its series of *Markov matrices* (or *impulse response matrices*, or *weighting matrices*) $K(j)$ $(j = 1, 2, \ldots)$ (recall that we assume that $K(0) = I$, which is in correspondence with the output equation in the system representation). In terms of the matrices A, B and C we find that $K(j) = CA^{j-1}B$ for $j = 1, 2, \ldots$. Using these, we can form the associated *block Hankel matrix* $\mathcal{H}$ as

$$\mathcal{H} = \begin{pmatrix} K(1) & K(2) & \cdots \\ K(2) & K(3) & \cdots \\ \vdots & \vdots & \end{pmatrix}$$

This matrix has finite rank n, and is the basis for many constructions in systems theory, such as for instance the construction of the *overlapping state-space parametrizations* that we shall use.
First of all, we notice that there might exist several triples (A, B, C) leading to the *same* transfer function $k(z)$. One can define an *equivalence relation* on our set of state-space systems by calling two systems *i/o-equivalent* if they correspond to the same transfer function. It is a known result that the equivalence class of a system (A, B, C) from our set above can be written as

$$\{(\tilde{A}, \tilde{B}, \tilde{C}) | \tilde{A} = TAT^{-1}, \tilde{B} = TB, \tilde{C} = CT^{-1}; T \text{ non-singular}, n \times n\}$$

Actually, we are interested in the equivalence classes rather than all their representatives. One way to select a representative from each equivalence class is by means of so-called canonical forms.
To obtain a canonical state-space realization for a given causal, rational transfer function $k(z)$ one can proceed by first choosing a *structure index* with respect to a *nice selection* for a basis for the row

space of $\mathcal{H}$. Depending on the structure index at hand, one can obtain various different minimal state–space representations in the model set under consideration. However, it is a known fact that in the multivariable case ($p > 1$) it is *impossible* to cover the whole model set with just one continuous canonical form. See Hazewinkel [21]. Therefore, one is forced to use several of them anyway. It turns out that the state–space approach towards realization leads in a quite natural way to *overlapping* parametrizations, as opposed to for instance the *ARMAX* systems approach. Detailed accounts of the construction procedure for overlapping state–space parametrizations can be found e.g. in Van Overbeek and Ljung [34,35], Picci [38], Glover and Willems [13] and Hannan and Deistler [15].

One can regard the set of equivalence classes of systems under consideration as a $2np$–dimensional differentiable manifold embedded in Hilbert space. For this, one could start by first constructing the Hilbert space of sequences $\{K(j)|j = 1, 2, \ldots\}$ satisfying

$$\text{tr}\left\{\sum_{j=1}^{\infty} K(j)^T K(j)\right\} < \infty$$

where $\text{tr}\{\cdot\}$ denotes the trace–operator. One then observes that (1) the sequences of Markov matrices corresponding to our model set are elements of the Hilbert space above; (2) one has an inner product on the Hilbert space above via

$$\langle k, \tilde{k}\rangle = \text{tr}\left\{\sum_{j=1}^{\infty} K(j)^T \tilde{K}(j)\right\}$$

where k, $\tilde{k}$ correspond to transfer functions in the model set and $K(j)$, $\tilde{K}(j)$ are their respective Laurent series coefficients in an obvious way. See e.g. Hazewinkel [21], Hazewinkel and Kalman [22], Hanzon [17,18,19]. The inner product defined above is the ℓ^2–inner product, and alternative definitions, leading to alternative inner products (for instance via different weighting coefficients) are possible.
One now can make the manifold into a *Riemannian* manifold because an inner product on the embedding Hilbert space induces in a natural way a Riemannian metric on the system manifold, since it defines an inner product on the tangent spaces to the manifold in a continuous and natural way. By choosing an appropriate norm on the embedding Hilbert space one can express his intuition and views about when two systems exhibit closely related behaviour and when not. Such considerations might be translated into different weighting coefficients with respect to the impulse response matrices, yielding different norms. A helpful instrument in this approach is that the difference of two systems (considered as i/o–mappings) can itself again be considered a linear finite dimensional system. See Hanzon [18,19]. However, also alternative approaches are possible, for example via different embeddings. Moreover, it turns out to be possible as well to use the *asymptotic, average Fisher information per observation* (in a prediction error setting) for the definition of a Riemannian metric. See also Peeters [37].

Summarizing, we have indicated the following.
(1) The model set under consideration is quite general; only minor restrictions are made in case stationary processes (without trends and harmonic components) are considered. The only serious drawback is in the choice of a fixed, prespecified order n. The assumption that the inputs are Gaussian is not essential so far, but will turn out to be convenient later.
(2) In identification, the need for a structure selection procedure is overcome by the use of overlapping parametrizations, as a switching strategy makes it possible to determine an appropriate structure on–line (cf. Van Overbeek and Ljung [34,35]).
(3) The model set under consideration corresponds to a differentiable manifold of dimension $2np$. It can be embedded into Hilbert space in various ways, thus inheriting several Riemannian metrics. Alternative Riemannian metrics can be constructed also, e.g. by means of the Fisher information matrix (in a prediction error setting).

2.2 Off–Line Prediction Error Identification

In this subsection we describe an approach for the identification of a linear system from the model set introduced before on the basis of a sample record of T datapoints $\{y(1), \ldots, y(T)\}$. These datapoints are assumed to belong to a realization of a linearly regular stationary process with rational spectral density of order n. The innovations $\epsilon(t)$ are assumed to be Gaussian distributed with covariance $\Sigma_\epsilon > 0$. In principle Σ_ϵ will be unknown, but this will hardly complicate the problem as we will see. To a model from the model set, given by the triple (A, B, C), corresponds a *representation of the innovations*, that can be written as

$$\begin{cases} x(t+1) = (A - BC)x(t) + By(t) \\ \epsilon(t) = -Cx(t) + y(t) \end{cases} \qquad (t \in \mathbf{Z})$$

This can be interpreted as the *inverse* system, describing the *o/i-mapping*. From this representation we can derive immediately a corresponding linear predictor, identical to the *steady-state Kalman filter* based on (A, B, C)

$$\begin{cases} \hat{x}(t+1) = (A - BC)\hat{x}(t) + By(t) \\ \hat{y}(t) = -C\hat{x}(t) \end{cases} \qquad (t = 1, 2, \ldots, T)$$

This leads to a corresponding linear system, which we call the *prediction error filter*, from which the prediction errors are obtained

$$\begin{cases} \hat{x}(t+1) = (A - BC)\hat{x}(t) + By(t) \\ e(t) = -C\hat{x}(t) + y(t) \end{cases} \qquad (t = 1, 2, \ldots, T)$$

with initial state $\hat{x}(1) = 0$.
From the optimality properties of the Kalman filter it follows that if $(A(\theta_*), B(\theta_*), C(\theta_*))$ denotes the *true* underlying system belonging to process $y(t)$, then (for T tending to infinity) the sequence of prediction errors $e(t)$ generated by the associated prediction error filter minimizes the following *prediction error criterion*

$$V(\theta) = \lim_{T \to \infty} \frac{1}{2T} \sum_{t=1}^{T} \|e(t, \theta)\|^2$$

over all θ such that $(A(\theta), B(\theta), C(\theta))$ belongs to the model set, where θ denotes a parameter vector of dimension $2np$ if we are using our overlapping parametrizations and $e(t, \theta)$ the prediction error for time t resulting from the prediction error filter based on θ.
Therefore, if one is interested in (one–step–ahead) prediction with respect to the data $y(t)$, $(t = 1, 2, \ldots, T)$, a reasonable strategy to arrive at an adequate description of the data seems to be to find a matrix triple $(\bar{A}, \bar{B}, \bar{C})$ in our model set, for which the corresponding prediction error filter yields prediction errors with a minimal total sum of squared lengths. In terms of a parametrization with θ the problem is to find $\bar{\theta}$ such that

$$V_T(\theta) = \frac{1}{2} \sum_{t=1}^{T} \|e(t, \theta)\|^2$$

is minimized for $\theta = \bar{\theta}$.

Some general remarks are in order.
(1) As pointed out for instance in Ljung [29] prediction error criteria and linear models (as used above) make sense also *without* a statistical framework. One needs to be aware that what is the "best" model in a given situation depends on many aspects, such as the intended use of a model, ease of computation, required accuracies, etc. The use of a *linear* model can then be justified depending on the situation at hand an the use of a prediction error criterion on the intended applications.
(2) The prediction error criterion used above is just one from a large class of possibilities. See e.g. Ljung [29] or Söderström and Stoica [41] for a more general account. Also, there are alternative ways to arrive at the same prediction error criterion, e.g. via the principle of maximum likelihood.

(3) It should be noticed that indeed the covariance matrix Σ_ϵ did not play a role in the foregoing problem formulation. If Σ_ϵ is known, however, its inverse can be used as a weighting matrix in the prediction error criterion, leading to probably better results.
(4) The above *identification problem* (i.e. the problem of finding an appropriate model in a chosen model class, based on available measurement data) belongs to the class of *off-line* or *batch* identification, since we assume *all* datapoints to be available from the beginning. We are not in the situation where a stream of new measurements is coming in, on basis of which a current estimated model has to be improved all the time.
(5) Notice that the criterion $V_T(\theta)$ belongs to the class of nonlinear least squares criteria, which follows easily if one stacks all T prediction error vectors into one big vector of dimension pT.
(6) According to the foregoing remark the method of Gauss–Newton is applicable as a method for finding a solution to the identification problem. Here we must notice that although we are *not* in one of the two cases mentioned in the previous section for which the Gauss–Newton matrix and the Hessian coincide, we still expect the Gauss–Newton method to exhibit excellent local convergence behaviour for *large* values of T, since the term in the Hessian neglected by Gauss–Newton has *zero expectation*. (Cf. Hanzon [16].)

We can now summarize the identification problem under consideration.

> We start from a record of T datapoints $y(t)$, which are assumed to stem from a (linearly regular, full rank) stochastic process. We want to model this process by a linear model of prespecified order n (as described above). The model should be good in the sense that the prediction error filter based on it produces small prediction errors (on the average). For this we adopt the prediction error criterion $V_T(\theta)$ specified above. We notice that this is a nonlinear least squares criterion. We next notice that our model set can be regarded as a differentiable manifold. It can be covered with overlapping parametrizations, as described in Van Overbeek and Ljung [34,35]. On the manifold we can define various Riemannian metrics, expressing our ideas about distances between systems. The use of overlapping parametrizations together with a parametrization selection strategy prevents the necessity of preliminary structural identification. We are thus in the situation where we have to minimize a nonlinear least squares criterion over a Riemannian manifold, for which overlapping parametrizations are available. This can be done via the techniques described in Section 1. Of particular interest are the methods of Gauss–Newton and Levenberg–Marquardt, but also Riemannian steepest descent (in order to obtain more insight in the properties of the various Riemannian metrics and their relation to the prediction error identification problem).

We conclude this section by showing how the partial derivatives of the prediction errors with respect to the parameters can be obtained. The procedure is relatively straightforward: we can extend the prediction error filter by partial differentiation of all equations with respect to the parameters. We then get

$$\begin{cases} \hat{x}(t+1) = (A-BC)\hat{x}(t) + By(t) & \\ \frac{\partial \hat{x}}{\partial \theta^i}(t+1) = \frac{\partial (A-BC)}{\partial \theta^i}\hat{x}(t) + (A-BC)\frac{\partial \hat{x}}{\partial \theta^i} + \frac{\partial B}{\partial \theta^i}y(t) & (t = 1,\ldots,T) \\ e(t) = -C\hat{x}(t) + y(t) & (i = 1,\ldots,2np) \\ \frac{\partial e}{\partial \theta^i}(t) = -\frac{\partial C}{\partial \theta^i}\hat{x}(t) - C\frac{\partial \hat{x}}{\partial \theta^i}(t) & \end{cases}$$

with initial states $\hat{x}(1) = 0$ and $\frac{\partial \hat{x}}{\partial \theta^i}(1) = 0$, $(i = 1,\ldots,2np)$.
Thus, we see that the first order derivatives required by the Gauss–Newton and Levenberg–Marquardt methods can be obtained exactly. There is no approximation with respect to the criterion $V_T(\theta)$ involved, since by definition we start all filters with zero initial conditions.

3 Computer Experiments and Results

In order to illustrate the concepts and methods described in the previous sections, we shall discuss the results of some computer experiments. We have investigated in detail the off–line identification of a model for a sample of output data that was generated by computer simulation. The characteristics of the data generating process are

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0.3 & -0.1 & 0.4 & 0.0 \\ -0.3 & -0.2 & 0.0 & 0.3 \end{pmatrix} \quad B = \begin{pmatrix} -0.1 & -0.2 \\ 0.0 & -0.3 \\ 0.2 & 0.2 \\ 0.3 & 0.4 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \Sigma_\epsilon = \begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 1.0 \end{pmatrix}$$

Thus, the "true" parameter vector is given by

$$x_* = (0.3, -0.1, 0.4, 0.0, -0.3, -0.2, 0.0, 0.3, -0.1, 0.0, 0.2, 0.3, -0.2, -0.3, 0.2, 0.4)^T$$

Obviously, the order of the data generating process is $n = 4$, and there are 2 inputs and outputs: $p = 2$.
We have used this set–up to generate a sample of size $T = 2000$. The stochastic input was simulated as 2–dimensional, zero mean, *Gaussian* white noise with unit covariance (as specified above).
Another interesting aspect of the chosen data generating process is that within the class of parametrizations under consideration there is *only one* parameter chart by which it can be represented.
In total, for $n = 4$ and $p = 2$, there are 3 parameter charts. These correspond to the following structures with respect to (A, B, C):

Structure 1

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ * & * & * & * \\ 0 & 0 & 0 & 1 \\ * & * & * & * \end{pmatrix} \quad B = \begin{pmatrix} * & * \\ * & * \\ * & * \\ * & * \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Structure 2

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \quad B = \begin{pmatrix} * & * \\ * & * \\ * & * \\ * & * \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Structure 3

$$A = \begin{pmatrix} * & * & * & * \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ * & * & * & * \end{pmatrix} \quad B = \begin{pmatrix} * & * \\ * & * \\ * & * \\ * & * \end{pmatrix} \quad C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Here the stars denote parameter locations. Thus, for the data generating system we are only dealing with structure 2.
We have considered identification starting from 3 different starting points. These are given by

$$x_0(1) = (0.5, 0.0, 0.1, -0.1, -0.2, 0.1, -0.3, -0.1, 0.1, 0.2, -0.3, -0.1, 0.0, -0.2, 0.0, -0.2)^T$$

$$x_0(2) = (0.0, 0.1, 0.1, -0.1, -0.2, 0.1, 0.0, -0.1, 0.1, -0.2, 0.1, -0.1, 0.0, -0.1, 0.1, -0.2)^T$$

$$x_0(3) = (0.4, -0.2, 0.3, -0.1, -0.2, -0.1, -0.1, -0.1, 0.2, -0.2, 0.1, 0.1, -0.3, -0.2, 0.3, 0.1)^T$$

method	$x_0(1)$	$x_0(2)$	$x_0(3)$
1	36*	20	14
2	> 100	20	13
3	37*	19	15
4	19	21	15
5	29*	21	15
6	> 100	> 100	> 100
7	crash	> 100	> 100
8	31*	23	15

where $x_0(1)$ and $x_0(3)$ are in structure 2 (the same as the data generating system), and $x_0(2)$ is in structure 1.
The minimization methods under consideration all contain the on–line structure selecting strategy of Van Overbeek and Ljung. There are 8 methods we have studied:

1. Riemannian Levenberg–Marquardt method, with Riemannian metric derived from the ℓ^2–norm on Hilbert space and embedding as described in Section 2 (this is called the i/o–embedding).
2. Riemannian Levenberg–Marquardt method, with Riemannian metric derived from the ℓ^2–norm on Hilbert space but an alternative embedding taking the prediction error filters as our point of departure (this is called the o/i–embedding).
3. Riemannian Levenberg–Marquardt method, with Riemannian metric derived from the theoretical average asymptotical Fisher information per observation.
4. Standard Levenberg–Marquardt method, i.e. with the Euclidean metric on each parameter chart separately.
5. (Damped) Gauss–Newton method, where step–sizes are halved until a decrease in function value is obtained.
6. Riemannian gradient method, with Riemannian gradient as under 1. Here an initial step–size is obtained via optimization after quasi–linearization. When necessary, halving (as for Gauss–Newton) is applied.
7. Riemannian gradient method, with Riemannian metric as under 2, and choice of step–size as under 6.
8. Riemannian gradient method, with Riemannian metric as under 3, and choice of step–size as under 6.

The table above contains the numbers of iterations needed to reach a local minimum in each situation. The situations indicated by a star converged to a local minimum in structure 1 that is not a global minimum. The crash for method 7 (after 5 iterations) was due to the fact that instability occurred: the structure selection procedure requires stability of A, which was not automatically imposed because the Riemannian metric involved requires only stability of $A - BC$. (For the other Riemannian metrics stability of A was ensured implicitly.)
In the figures at the end of this section we show plots of some parameter estimates as well as the achieved criterion values. Also the structure indices are shown corresponding to the iterations (where method 1 is the lowest and method 8 the highest row in each plot). In plots 13 to 15 we show the *exact* criterion values $V(\theta)$, i.e. for $T \to \infty$, based on the estimated models, for data generated by the true underlying process.
Some remarkable features of the results that should be noticed are:

1. The third Riemannian metric (corresponding to the theoretical Fisher information) performs extraordinarily well in combination with a Riemannian gradient strategy. Compared to the

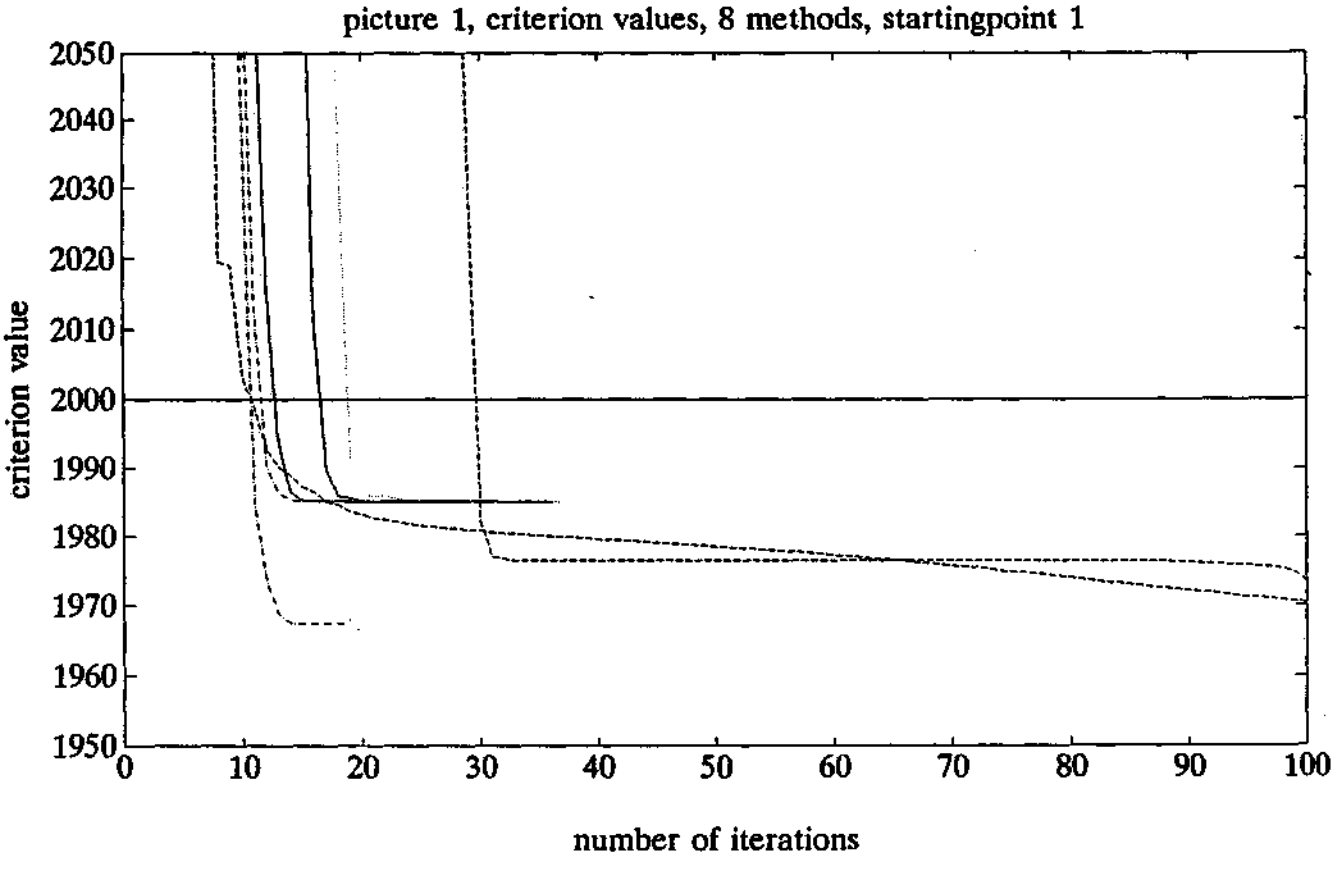
other two metrics under investigation its local convergence properties are significantly better. This can be explained by the fact that for $T \to \infty$ the Fisher information converges to the Gauss–Newton matrix (or conversely), whereas moreover the Hessian becomes equal to the Gauss–Newton matrix also. Thus, we expect local convergence that is *superlinear* (and in practice of almost *second order*). This is indeed confirmed by the experiments.

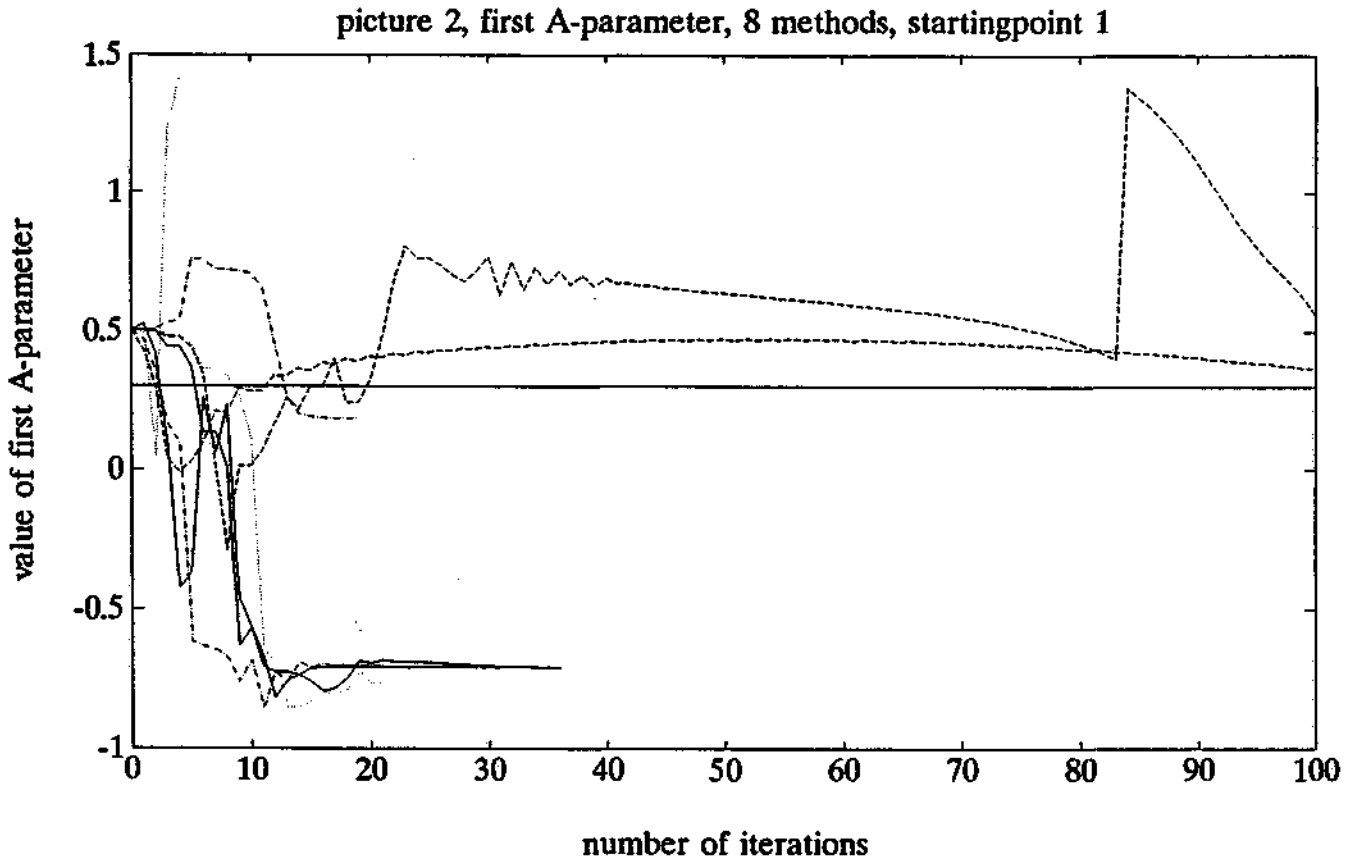
2. Starting point $x_0(3)$ is nearest to the "true" point x_*. This is clearly reflected by the fact that convergence in case 3 is quickest. On the other hand case 1 is furthest away. Here we see in many situations convergence to a strictly *local* minimum.

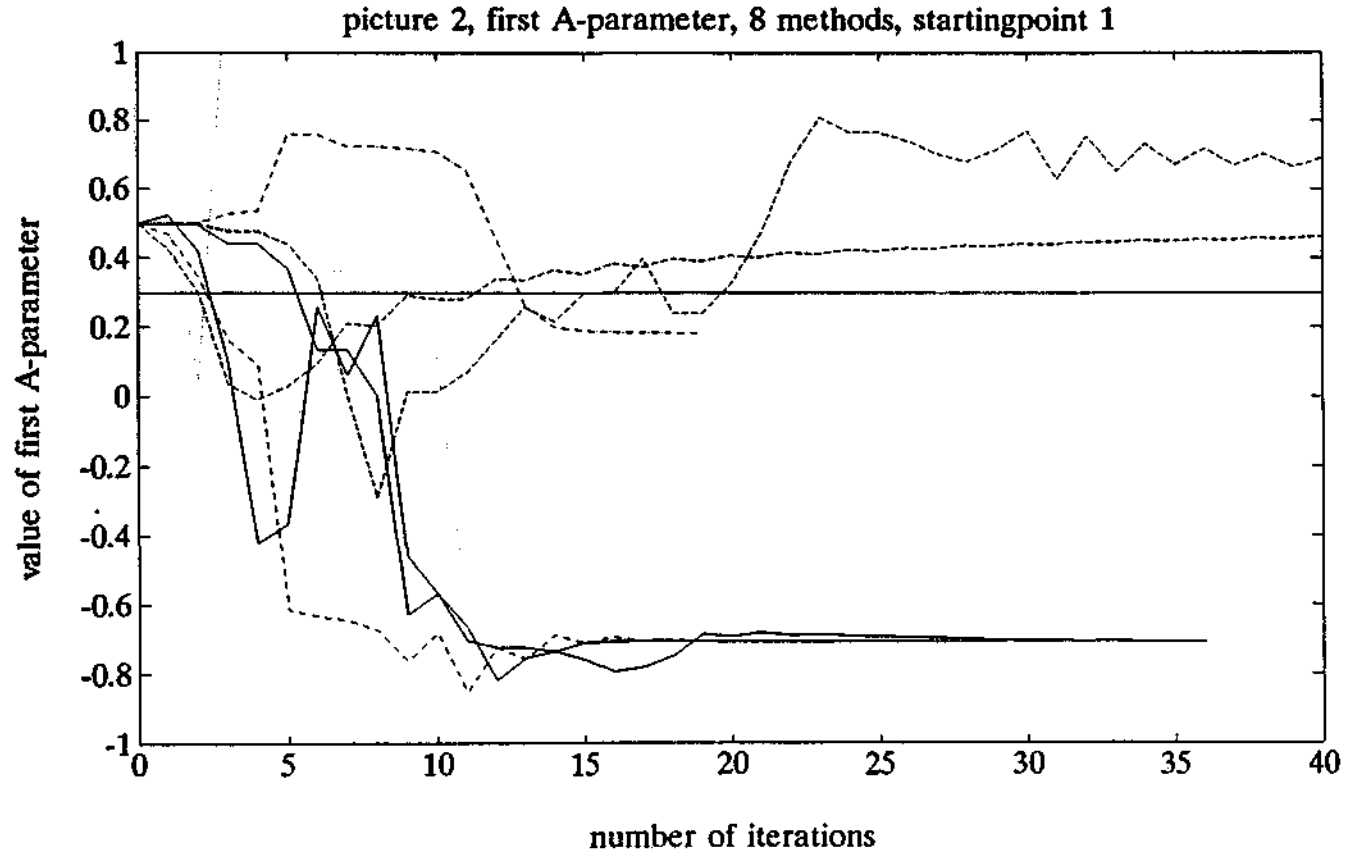
3. Convergence in the B-parameters usually occurs quickest, which is to be expected in some sense because they are weighted more than the other ones. We see that though the parameters can still be quite different and far from the true system, the resulting predictor can already be quite good.

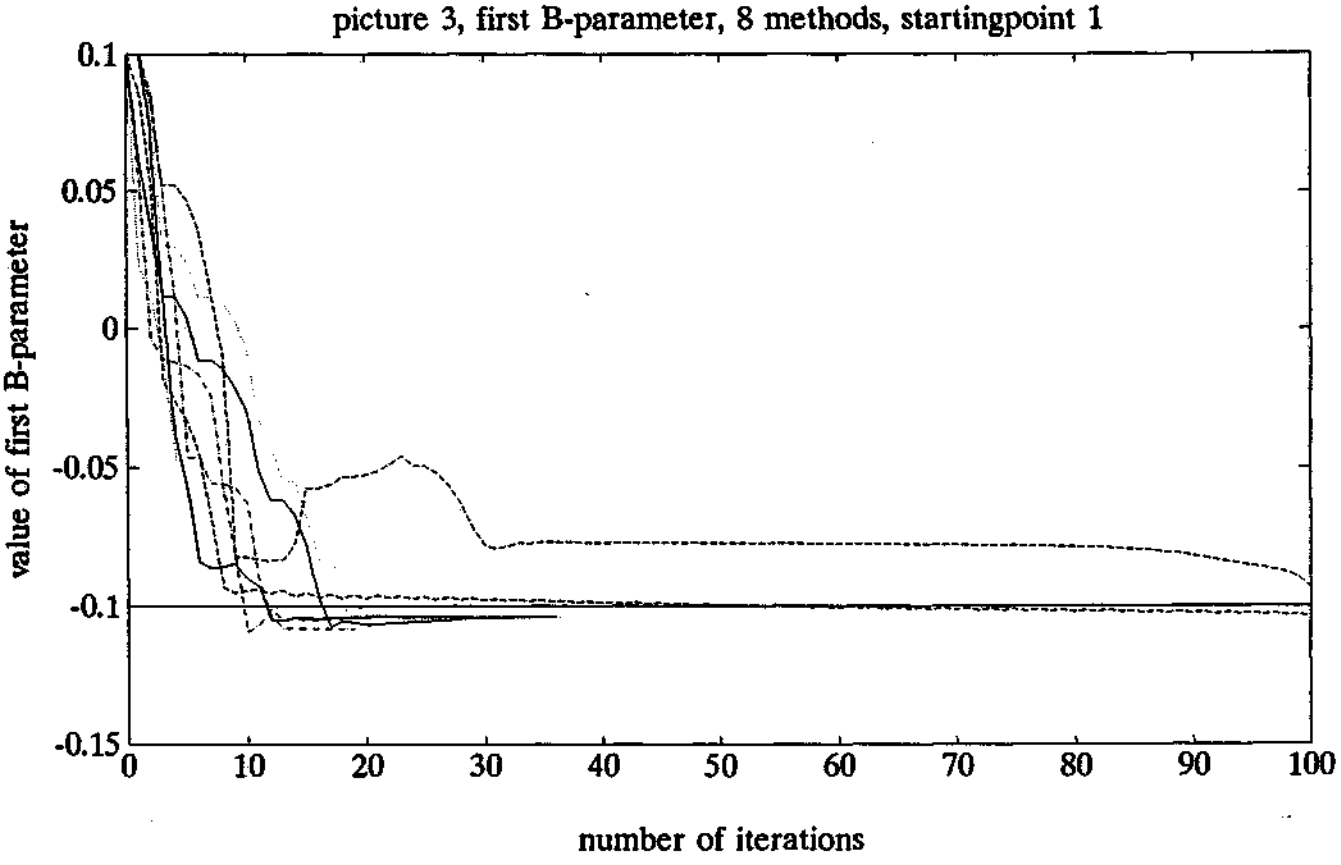
4. Notice the interesting behaviour of the structure selection and the effect it has on the convergence of parameter estimates. In particular we can see for method 2, starting point 1, that convergence is slow in structure 1. When eventually a switch to structure 2 occurs (after 83 iterations) speed of convergence is improved considerably. Notice that according to the results for starting points 2 and 3, there is no trouble with respect to speed of convergence near the optimum. Apparently the non–linearity aspects involved play an important role.

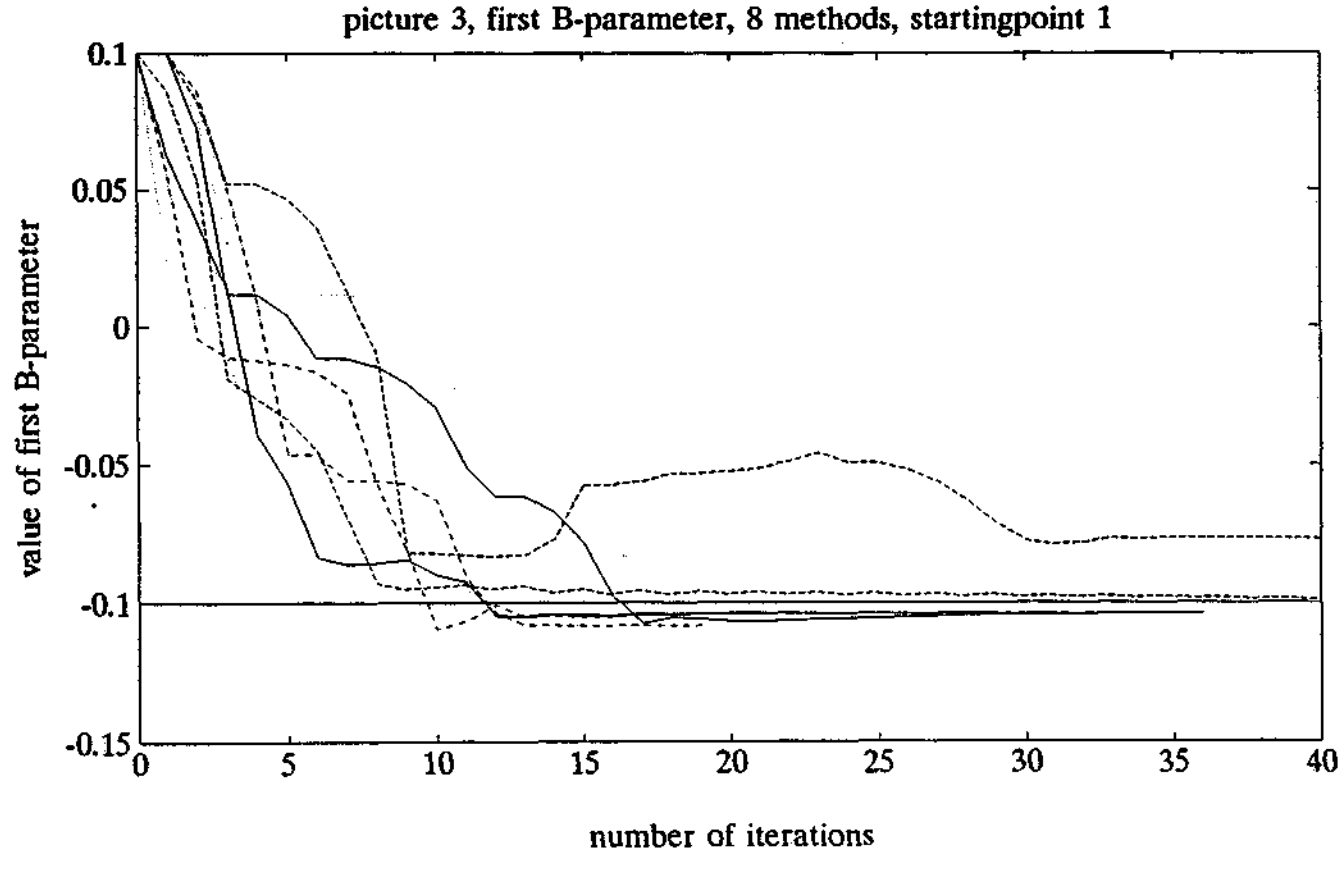
We conclude this paper by indicating further research (as to appear in Peeters [37]). This is directed towards the development and implementation of *recursive* methods for identification on Riemannian manifolds of systems. Also an extension to the situation where exogenous (deterministic, measured) inputs can be applied is studied.
Future research in this area should involve (a) investigation of which Riemannian metrics are especially suited both with respect to the behaviour of the identification algorithms and with respect to our notion of distance between systems; (b) the development of an on–line *order selection* procedure, because one of the main weak points of the current approach is the assumption that an order n is given; (c) broadening of the field of applications for which the ideas expressed in this paper could be useful.

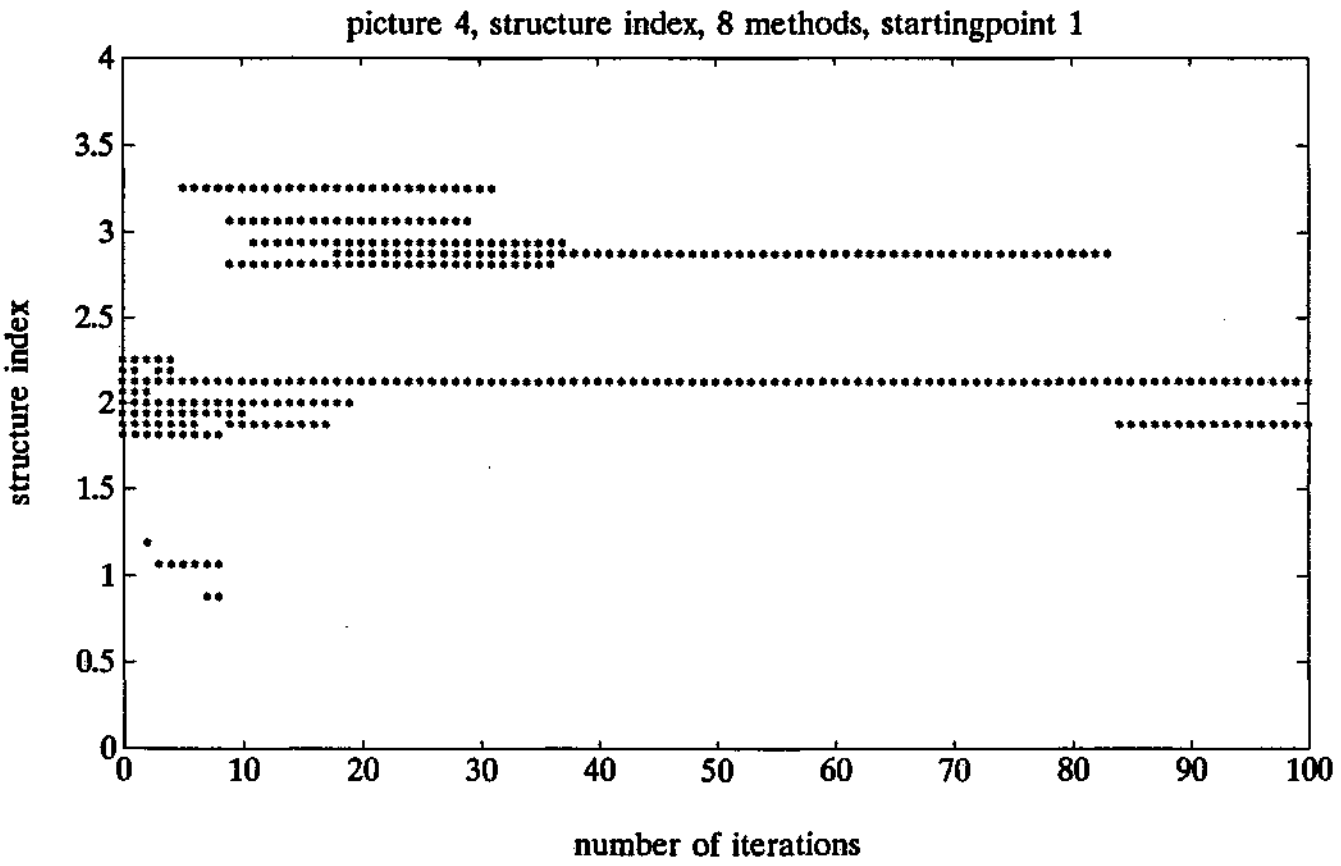

picture 1, criterion values, 8 methods, startingpoint 1
criterion value
2050
2040
2030
2020
2010
2000
1990
1980
1970
1960
1950
0
10
20
30
40
50
60
70
80
90
100
number of iterations

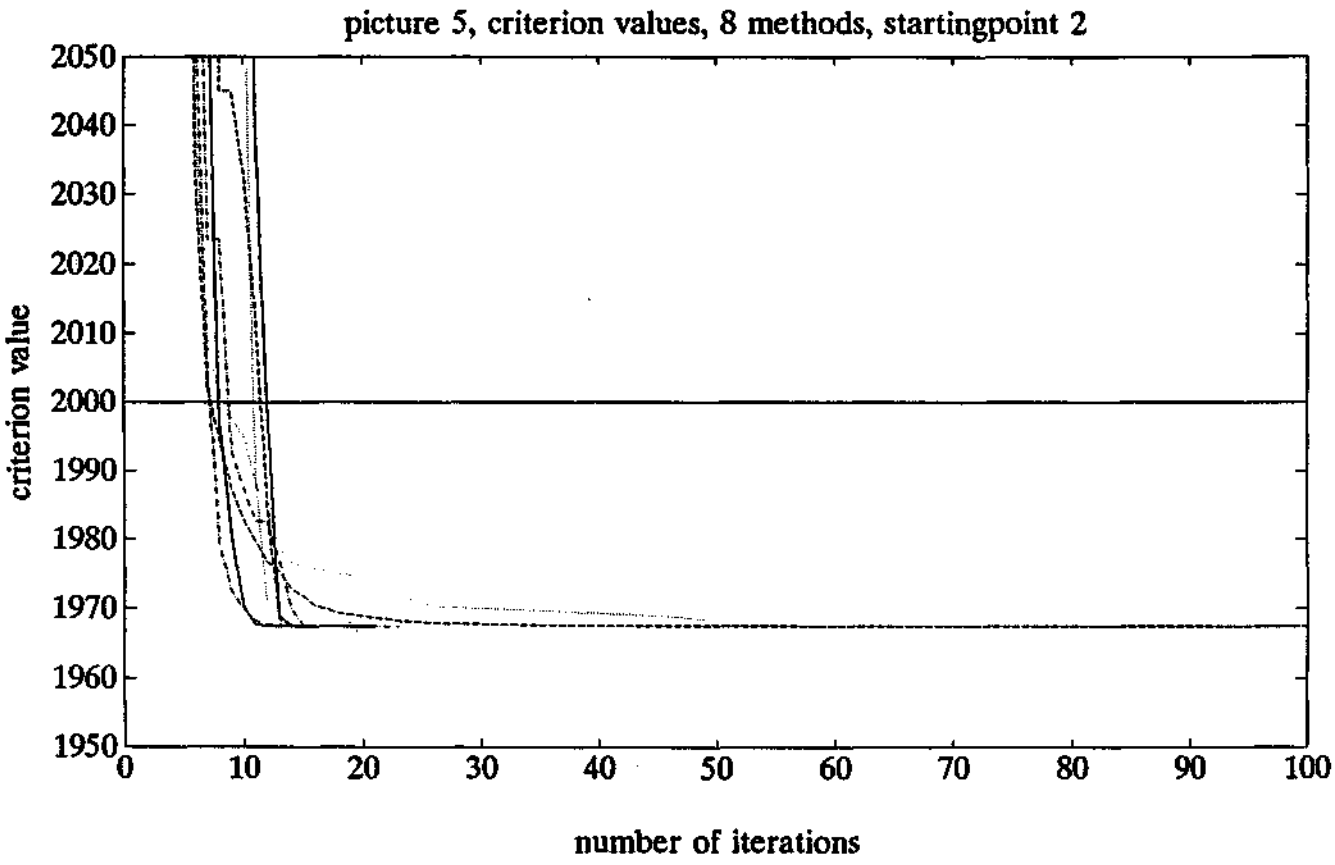

picture 2, first A-parameter, 8 methods, startingpoint 1
value of first A-parameter
number of iterations
1.5
1
0.5
0
-0.5
-1
0
10
20
30
40
50
60
70
80
90
100

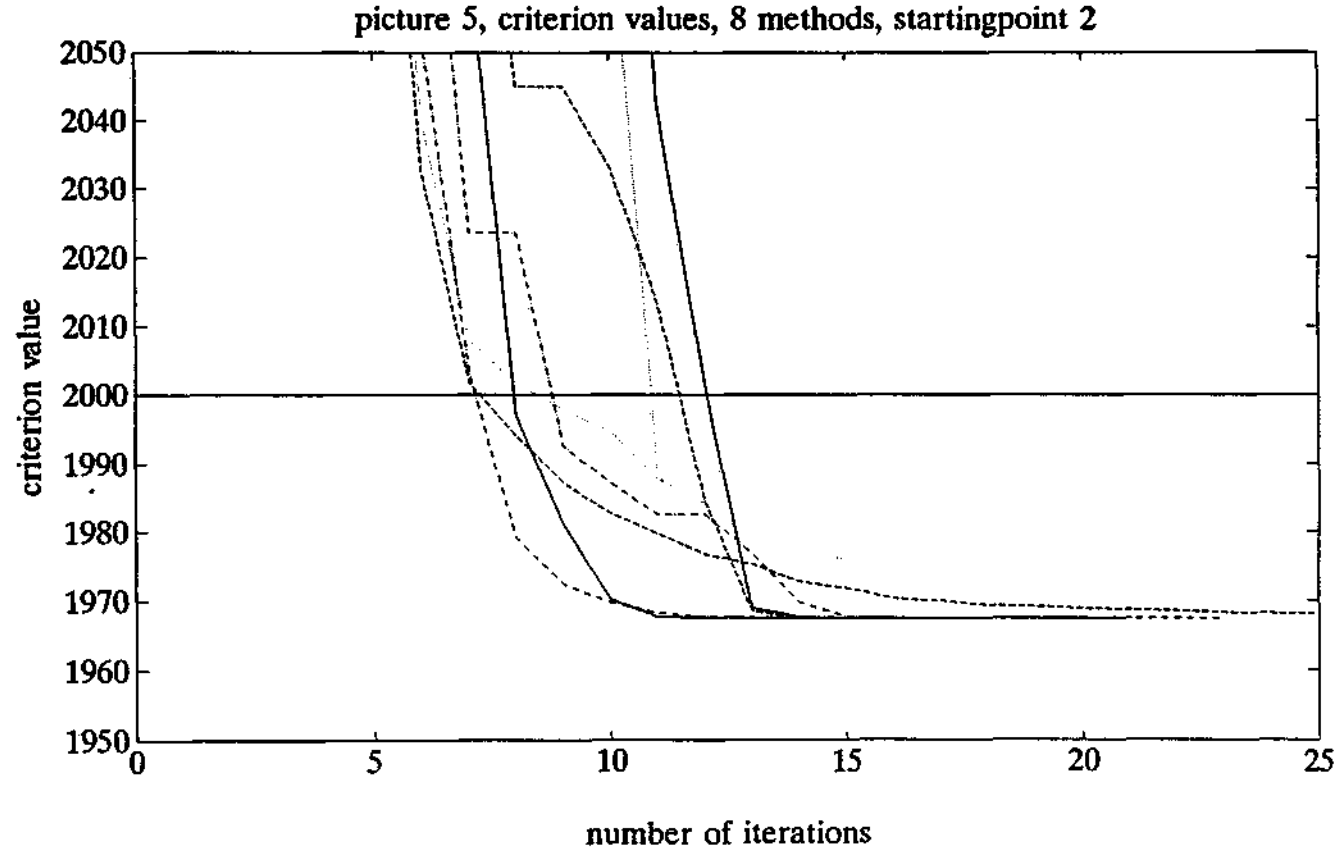

picture 2, first A-parameter, 8 methods, startingpoint 1
value of first A-parameter
number of iterations
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
5
10
15
20
25
30
35
40

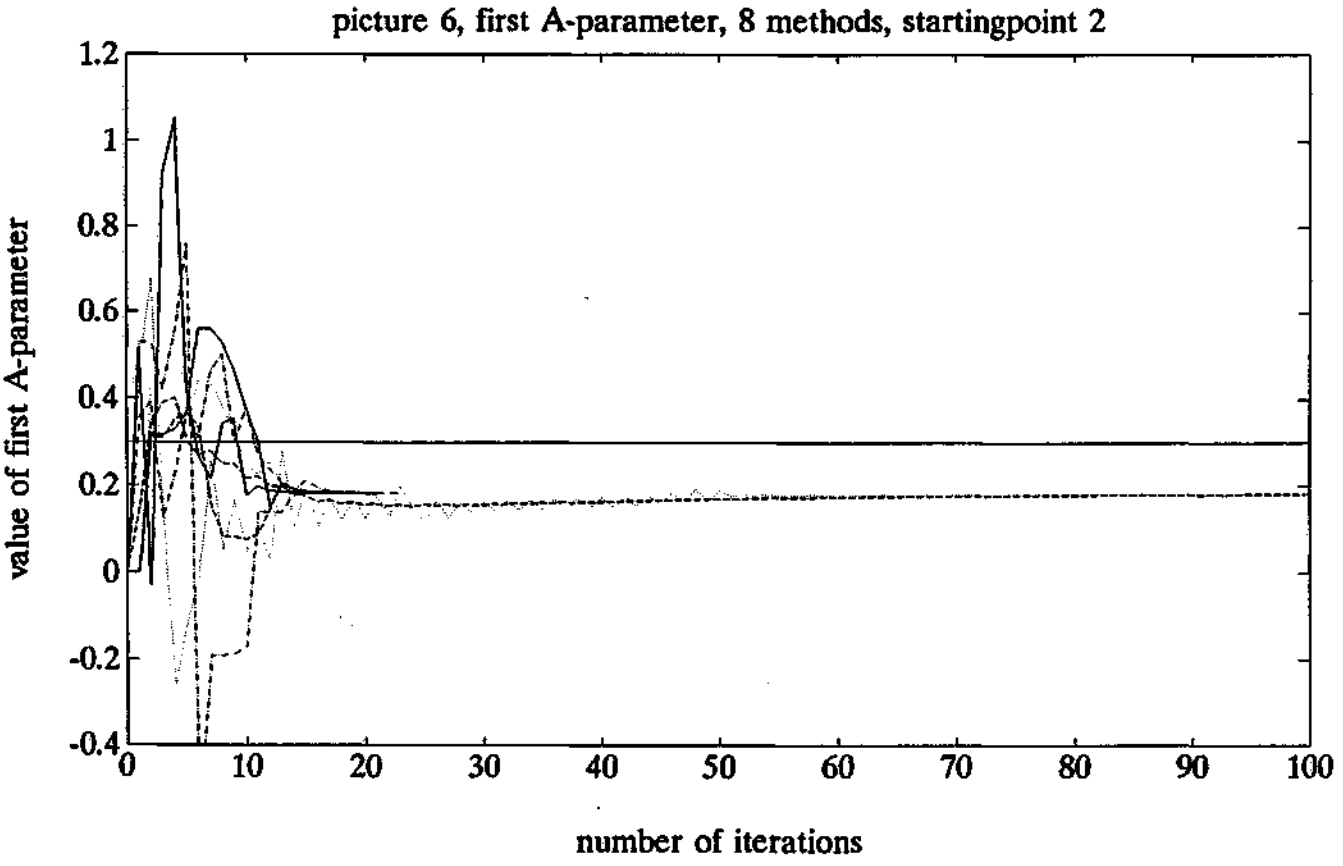

picture 3, first B-parameter, 8 methods, startingpoint 1
value of first B-parameter
number of iterations
0.1
0.05
0
-0.05
-0.1
-0.15
0
10
20
30
40
50
60
70
80
90
100

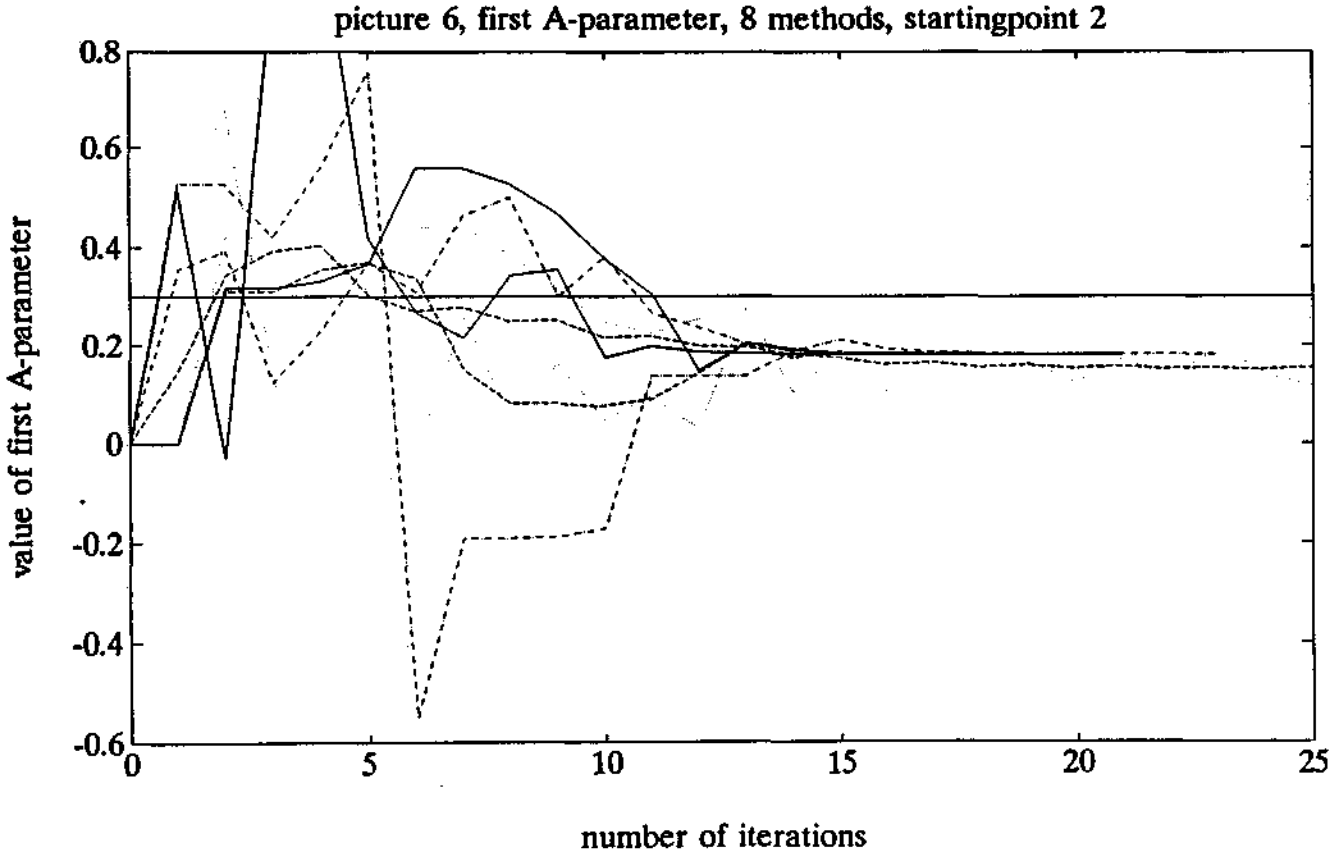

picture 3, first B-parameter, 8 methods, startingpoint 1
value of first B-parameter
number of iterations
0.1
0.05
0
-0.05
-0.1
-0.15
0
5
10
15
20
25
30
35
40

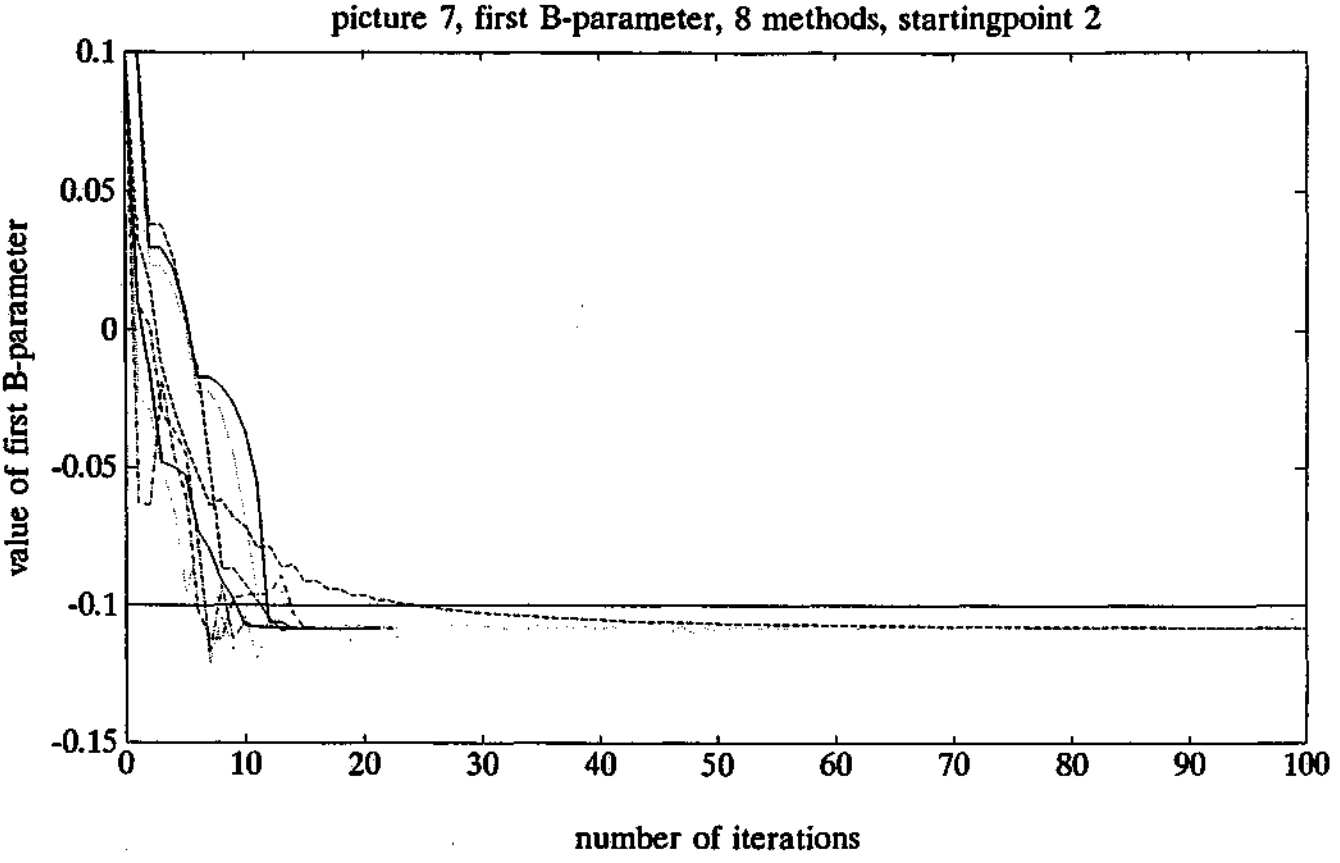

picture 4, structure index, 8 methods, startingpoint 1
structure index
number of iterations
0
0.5
1
1.5
2
2.5
3
3.5
4
0
10
20
30
40
50
60
70
80
90
100

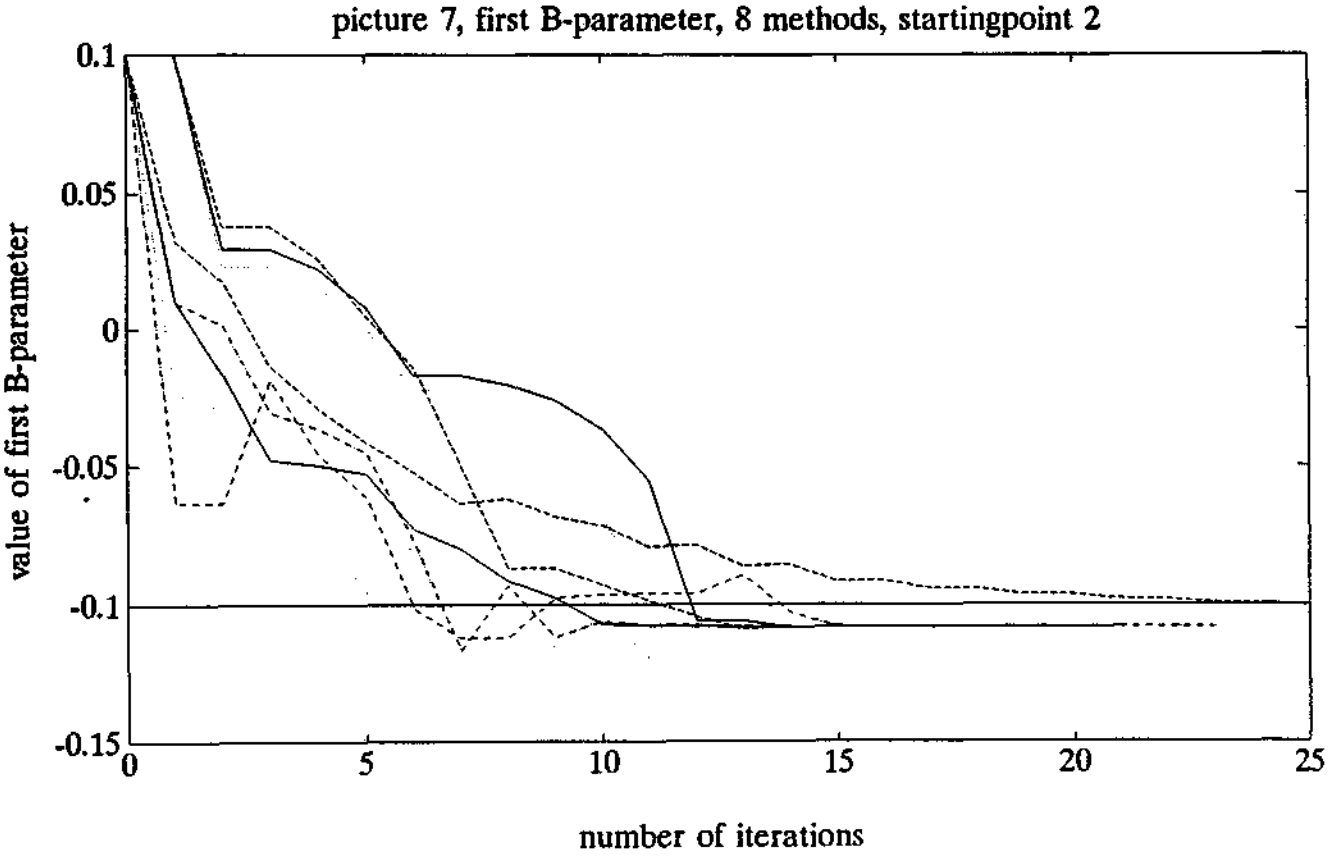

picture 5, criterion values, 8 methods, startingpoint 2
criterion value
2050
2040
2030
2020
2010
2000
1990
1980
1970
1960
1950
0
10
20
30
40
50
60
70
80
90
100
number of iterations

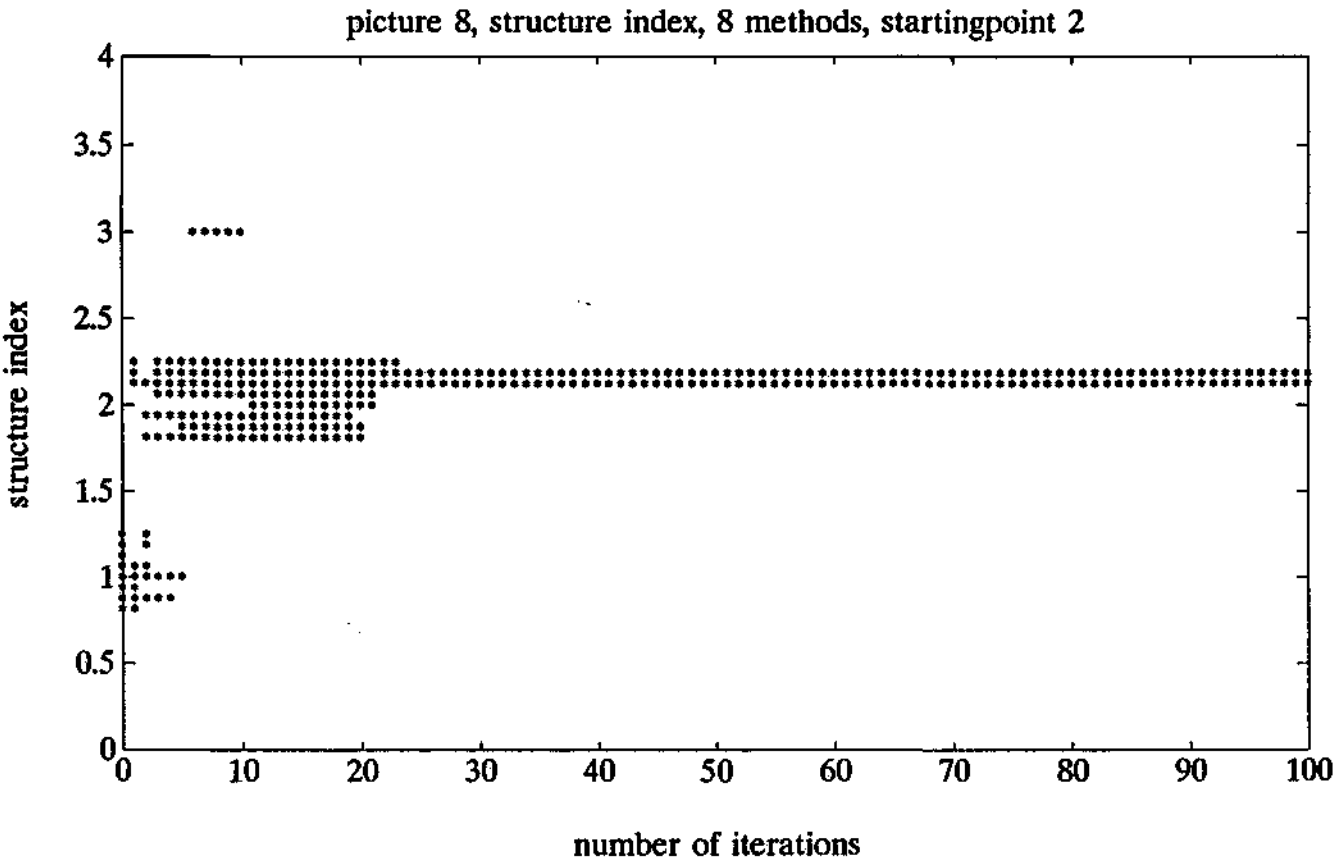

picture 5, criterion values, 8 methods, startingpoint 2
criterion value
2050
2040
2030
2020
2010
2000
1990
1980
1970
1960
1950
0
5
10
15
20
25
number of iterations

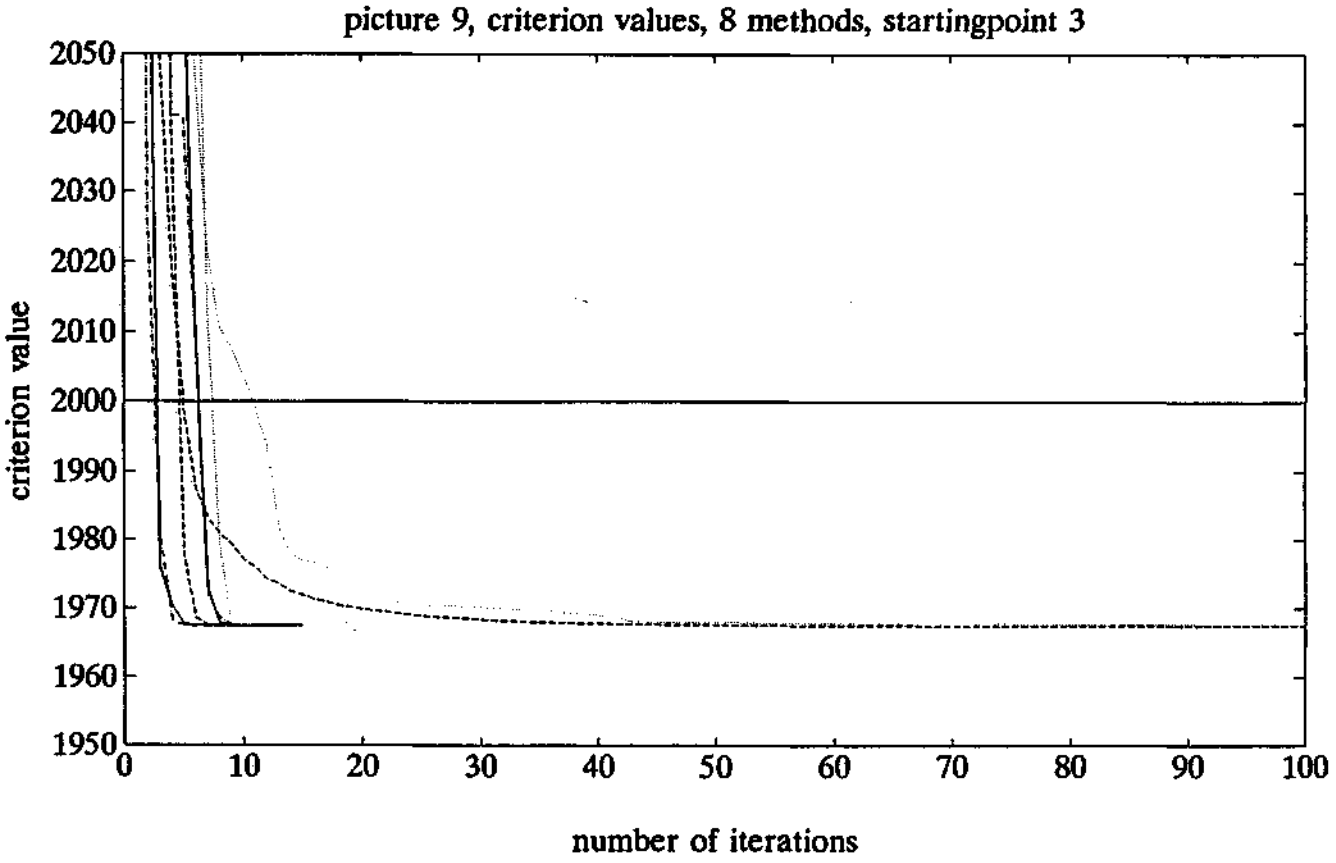

picture 6, first A-parameter, 8 methods, startingpoint 2
value of first A-parameter
number of iterations
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
0
10
20
30
40
50
60
70
80
90
100

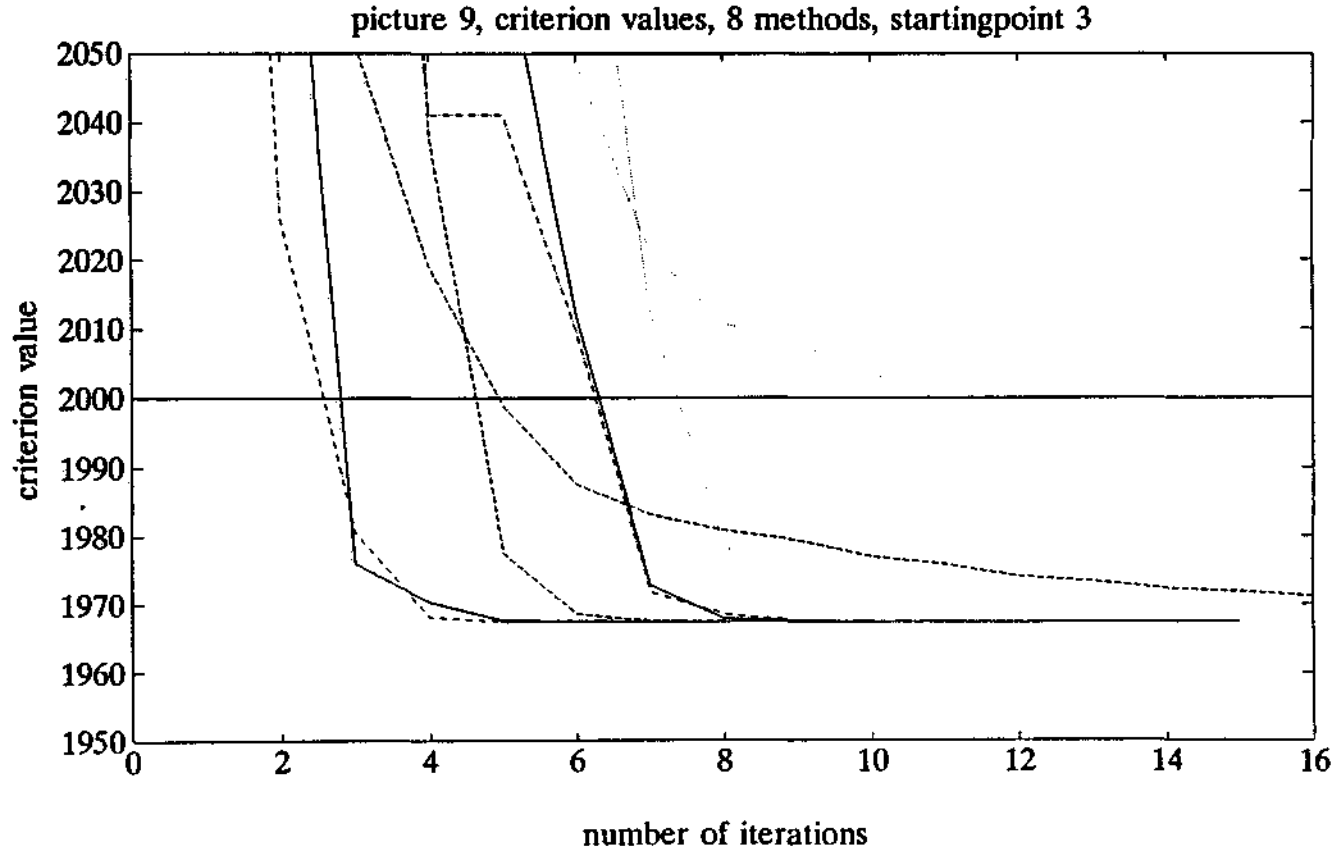

picture 6, first A-parameter, 8 methods, startingpoint 2
value of first A-parameter
number of iterations
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
0
5
10
15
20
25

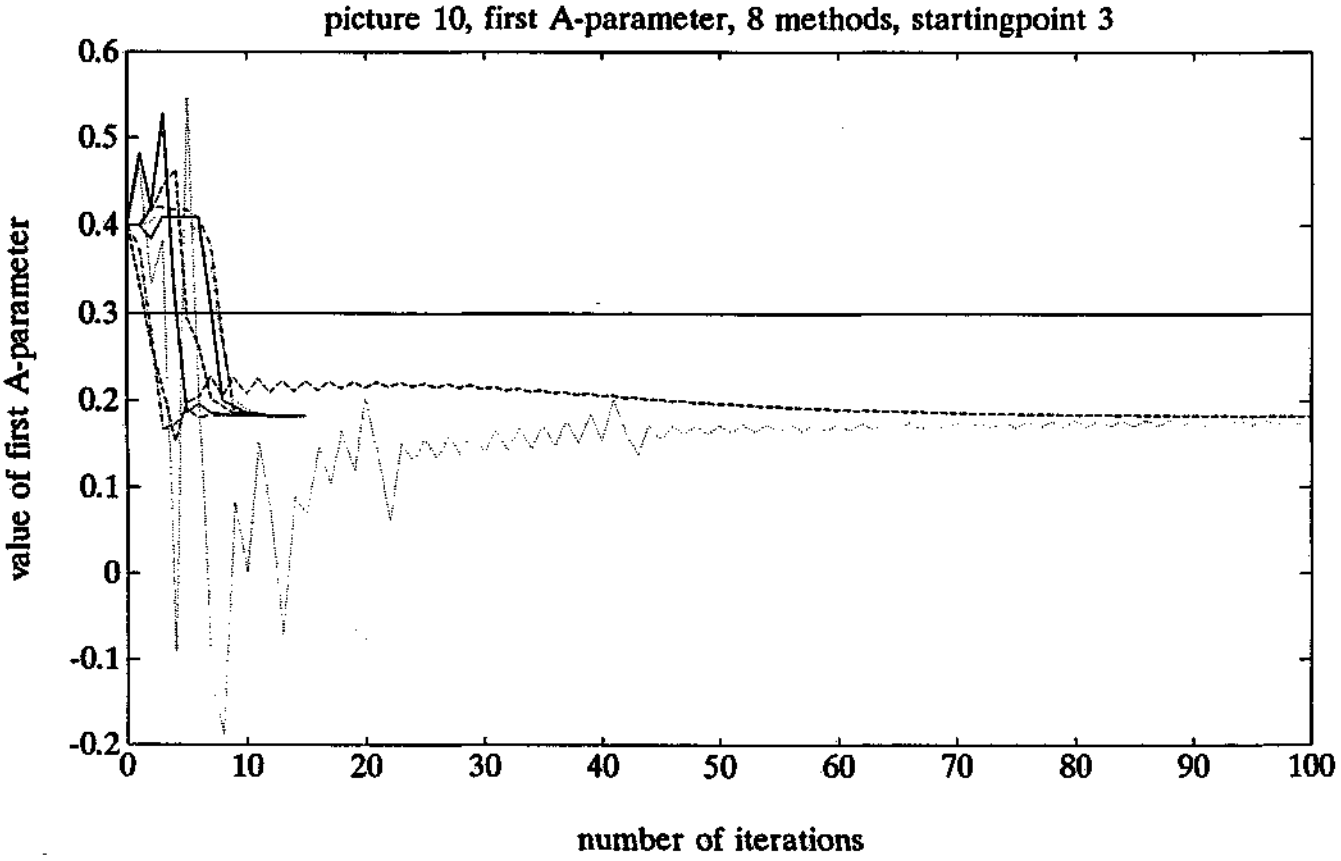

picture 7, first B-parameter, 8 methods, startingpoint 2
value of first B-parameter
0.1
0.05
0
-0.05
-0.1
-0.15
0
10
20
30
40
50
60
70
80
90
100
number of iterations

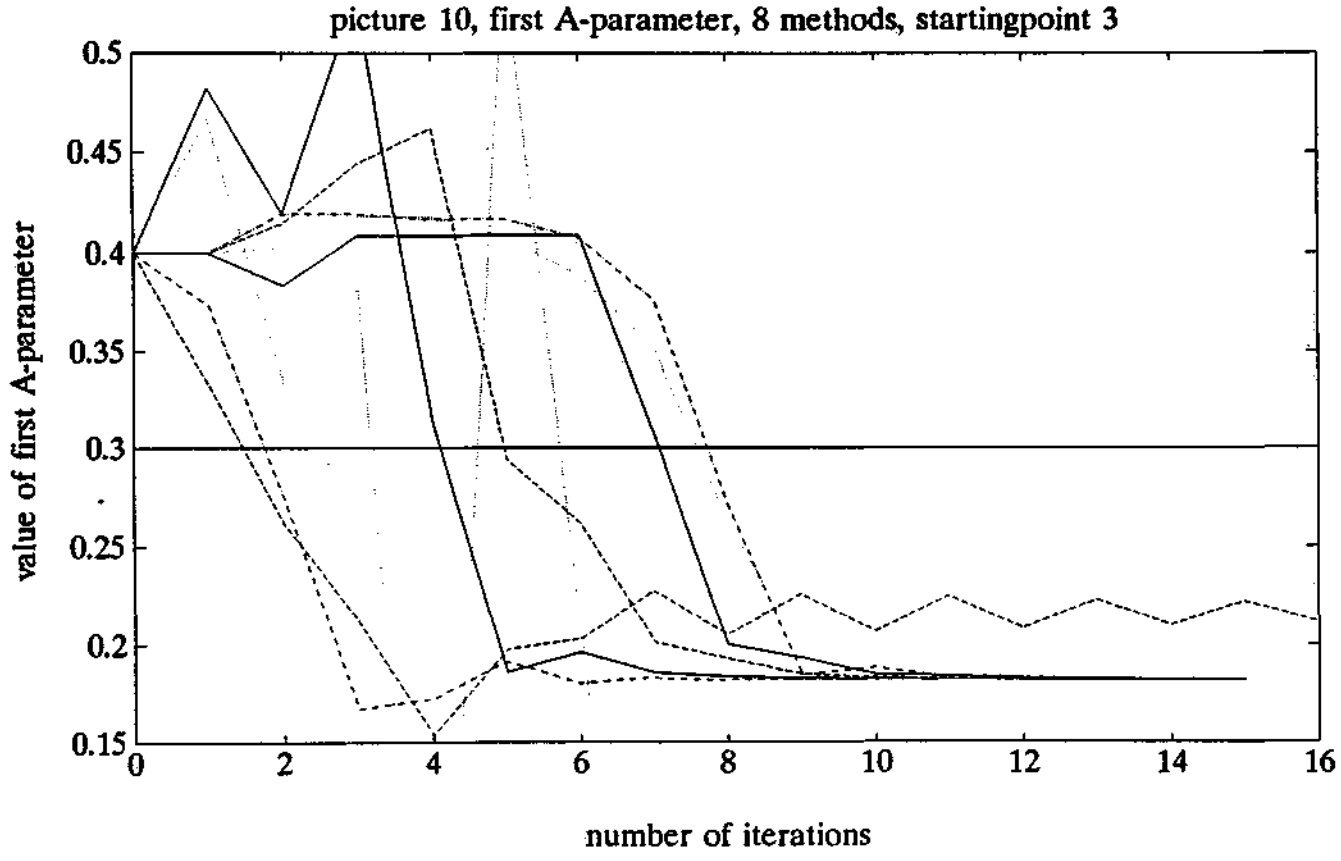

picture 7, first B-parameter, 8 methods, startingpoint 2
value of first B-parameter
0.1
0.05
0
-0.05
-0.1
-0.15
0
5
10
15
20
25
number of iterations

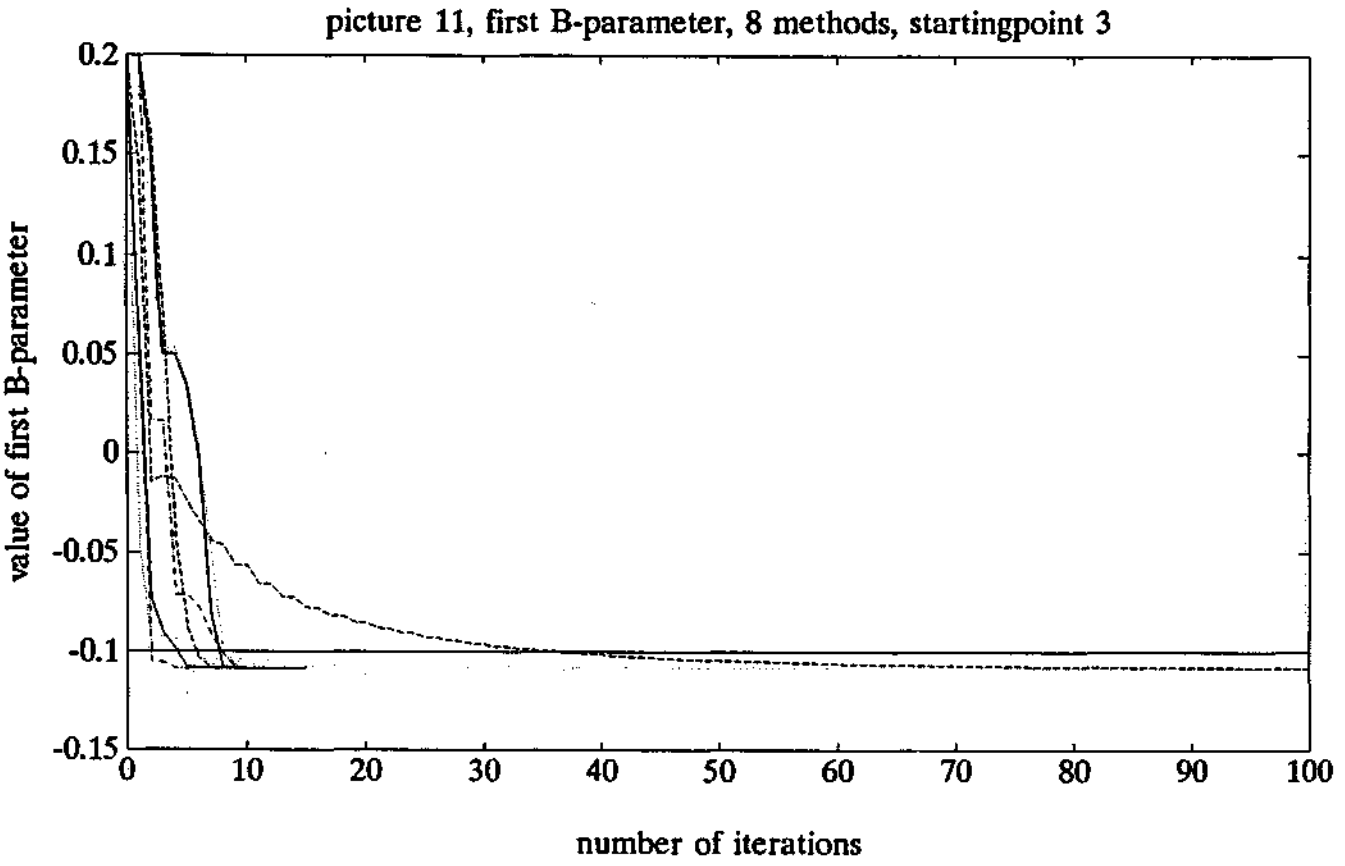

picture 8, structure index, 8 methods, startingpoint 2
structure index
number of iterations
0
0.5
1
1.5
2
2.5
3
3.5
4
0
10
20
30
40
50
60
70
80
90
100

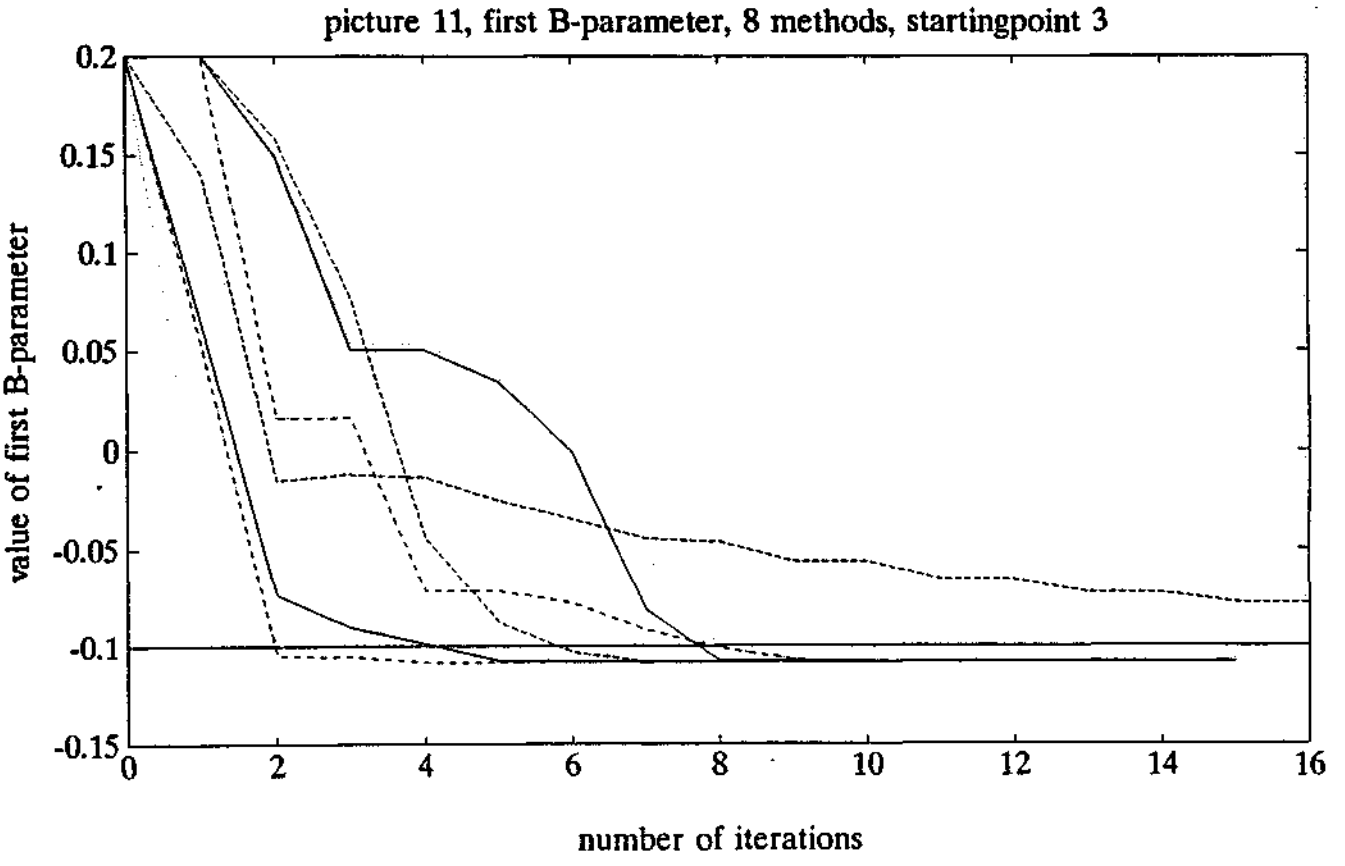

picture 9, criterion values, 8 methods, startingpoint 3
criterion value
2050
2040
2030
2020
2010
2000
1990
1980
1970
1960
1950
0
10
20
30
40
50
60
70
80
90
100
number of iterations

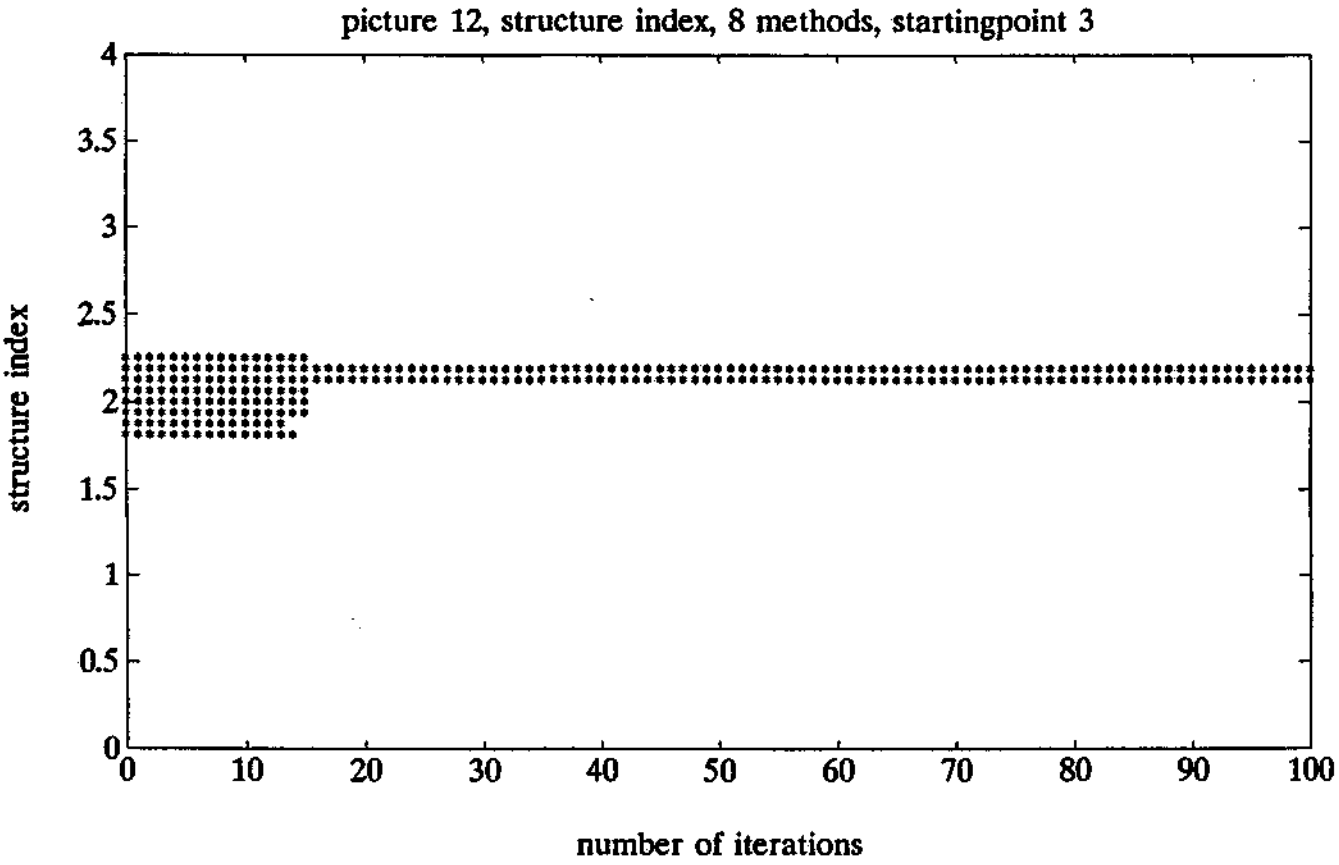

picture 9, criterion values, 8 methods, startingpoint 3
criterion value
2050
2040
2030
2020
2010
2000
1990
1980
1970
1960
1950
0
2
4
6
8
10
12
14
16
number of iterations


picture 10, first A-parameter, 8 methods, startingpoint 3
value of first A-parameter
number of iterations
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
0
10
20
30
40
50
60
70
80
90
100

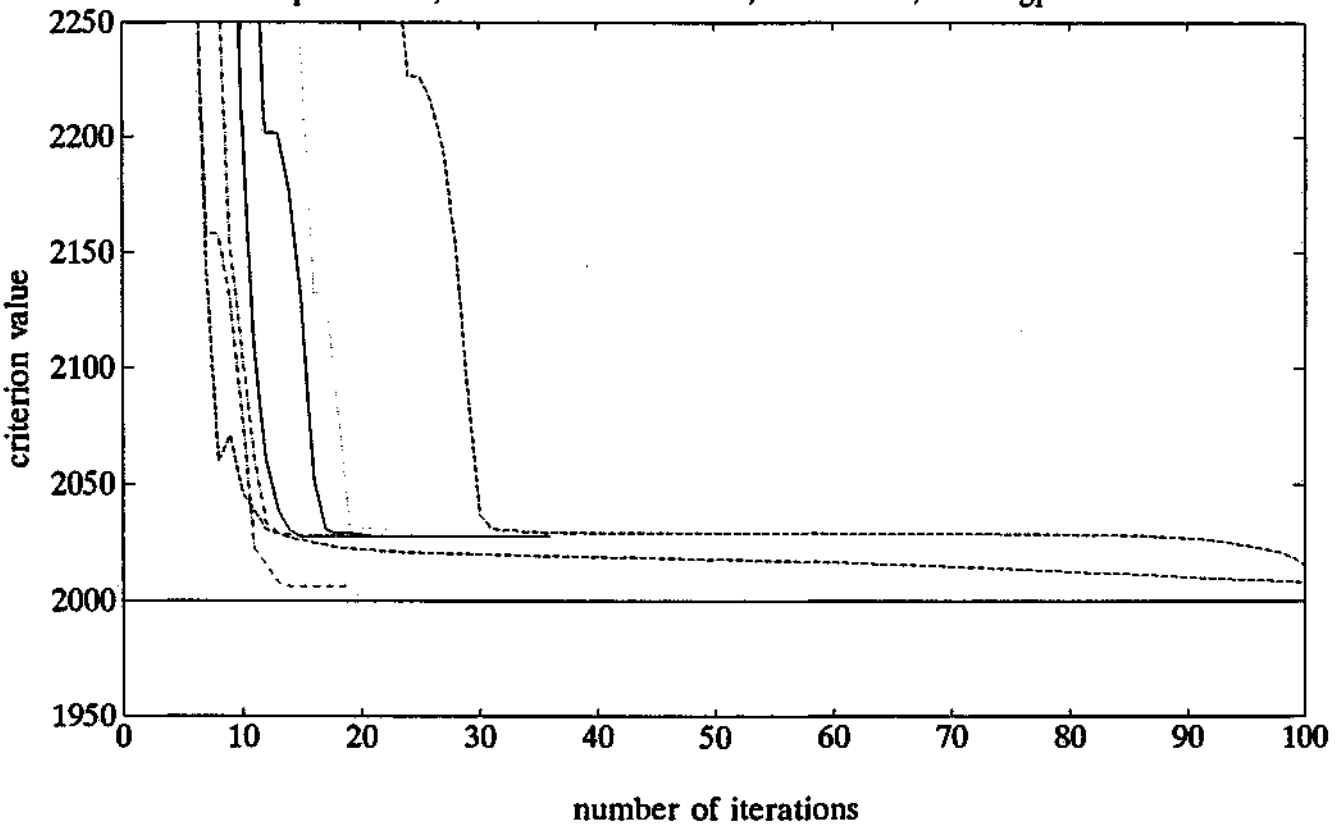

picture 10, first A-parameter, 8 methods, startingpoint 3
value of first A-parameter
number of iterations
0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0
2
4
6
8
10
12
14
16


picture 11, first B-parameter, 8 methods, startingpoint 3
value of first B-parameter
number of iterations
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
10
20
30
40
50
60
70
80
90
100

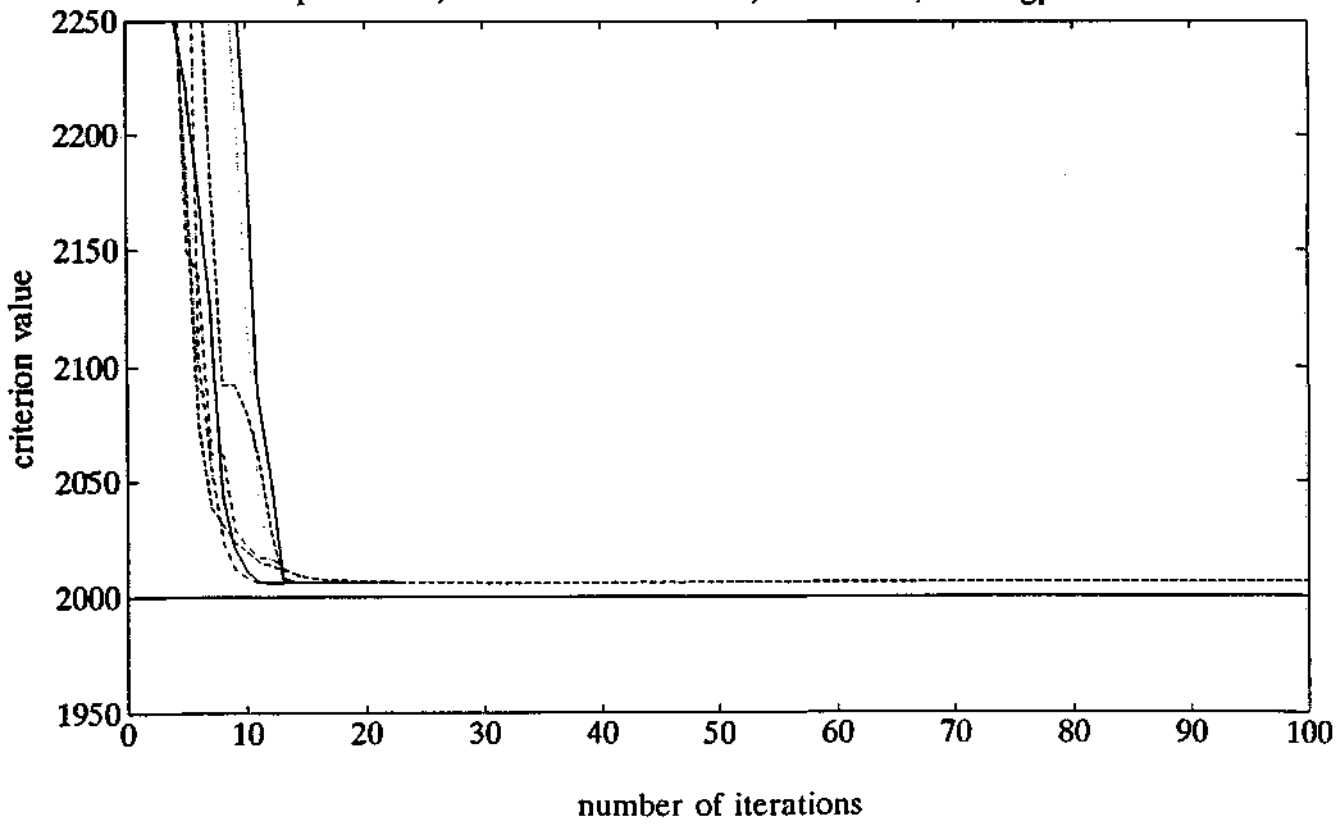

picture 11, first B-parameter, 8 methods, startingpoint 3
value of first B-parameter
number of iterations
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
0
2
4
6
8
10
12
14
16

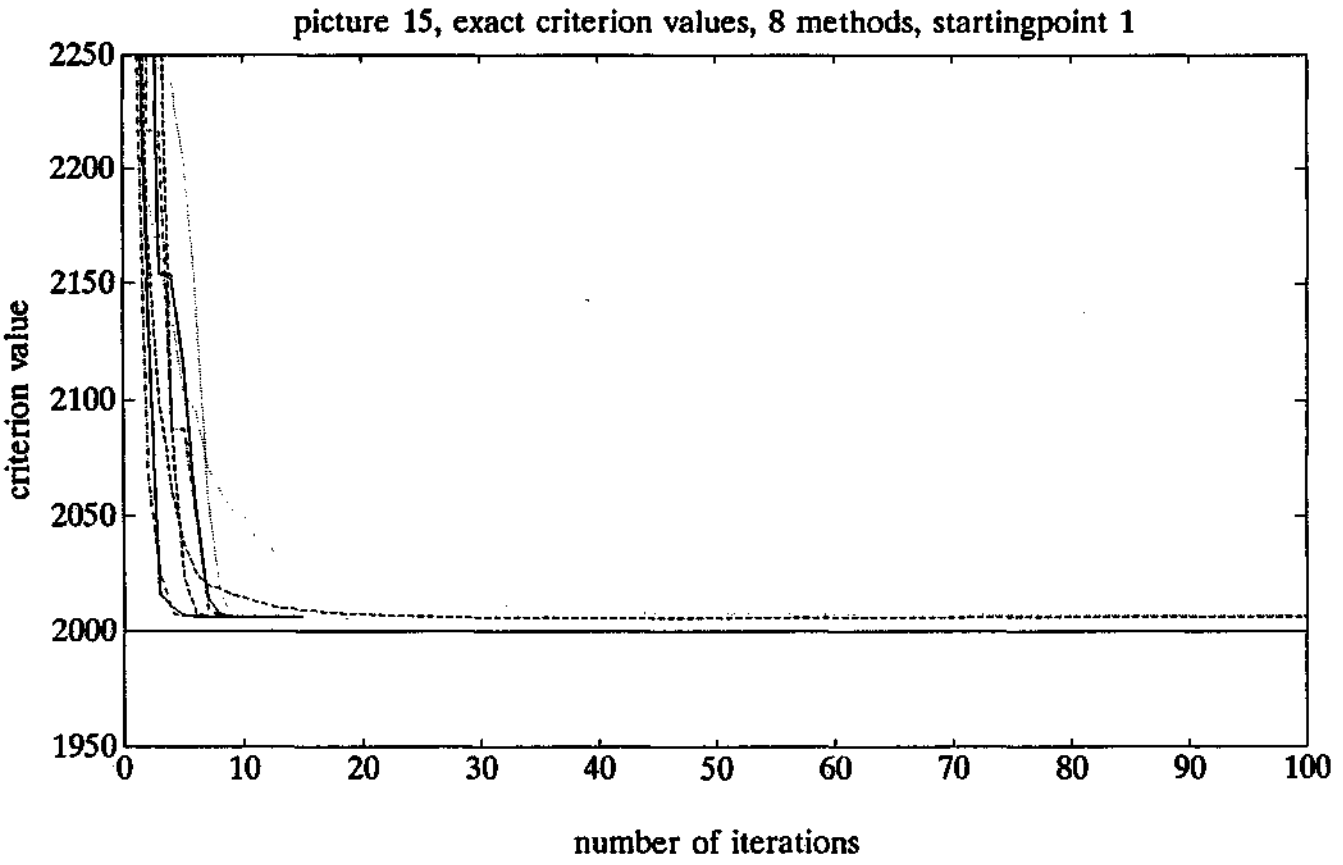

picture 12, structure index, 8 methods, startingpoint 3
structure index
number of iterations
0
0.5
1
1.5
2
2.5
3
3.5
4
0
10
20
30
40
50
60
70
80
90
100

picture 13, exact criterion values, 8 methods, startingpoint 1



picture 14, exact criterion values, 8 methods, startingpoint 1




picture 15, exact criterion values, 8 methods, startingpoint 1
criterion value
2250
2200
2150
2100
2050
2000
1950
0
10
20
30
40
50
60
70
80
90
100
number of iterations

References

[1] R.A. Abraham, J.E. Marsden, *Foundations of Mechanics* (2nd ed.). Reading, Mass.: Benjamin & Cummings, 1978.

[2] B.D.O. Anderson and J.B. Moore, *Optimal Filtering*. Englewood Cliffs: Prentice–Hall, 1979.

[3] Y. Bard, Comparison of gradient methods for the solution of nonlinear parameter estimation problems, *SIAM J. Num. Anal.* 7, 157–186, 1970.

[4] Y. Bard, *Nonlinear Parameter Estimation*. New York: Academic Press, 1974.

[5] G.J. Bierman, *Factorization Methods for Discrete Sequential Estimation*. New York: Academic Press, 1977.

[6] W.M. Boothby, *An Introduction to Differentiable Manifolds and Riemannian Geometry*. New York: Academic Press, 1975.

[7] J.E. Dennis, Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Englewood Cliffs: Prentice–Hall, 1983.

[8] R.A. Fisher, *Statistical Methods for Research Workers* (first edition 1925, eleventh edition 1950). Edinburgh: Oliver and Boyd, 1950.

[9] R.A. Fisher, *Contributions to Mathematical Statistics* (Collection of papers published 1920–1943). New York: Wiley, 1950.

[10] R. Fletcher, *Practical Methods of Optimization*, Vol. 1: Unconstrained Optimization. New York: John Wiley & Sons, 1980.

[11] D. Gabay, Minimizing a Differentiable Function over a Differentiable Manifold, *J. of Optimiz. Th. and Appl.* 37, 177–219, 1982.

[12] F.R. Gantmacher, *The Theory of Matrices*, Vol. I and II. New York: Chelsea, 1959.

[13] K. Glover and J.C. Willems, Parametrizations of Linear Dynamical Systems: Canonical Forms and Identifiability, *IEEE Trans. on Autom. Contr.*, Vol. AC-19, 640–646, 1974.

[14] E.J. Hannan, *Multiple Time Series*. New York: John Wiley & Sons, 1970.

[15] E.J. Hannan and M. Deistler, *The Statistical Theory of Linear Systems*. New York: John Wiley & Sons, 1988.

[16] B. Hanzon, On a Gauss–Newton identification method that uses overlapping parametrizations, *IFAC Identification and System Parameter Estimation 1985, York, UK*, 1671–1676, 1985.

[17] B. Hanzon, Riemannian geometry on families of linear systems, the deteministic case, Report 88-62, Delft University of Technology. Delft: Faculty of Mathematics and Informatics, 1988.

[18] B. Hanzon, *Identifiability, Recursive Identification and Spaces of Linear Dynamical Systems*, CWI Tracts 63, 64. Amsterdam: Centre for Mathematics and Computer Science, 1989.

[19] B. Hanzon, On the differentiable manifold of fixed order stable linear systems, *Systems & Control Letters* 13, 345–352, 1989.

[20] B. Hanzon and R.L.M. Peeters, On the Riemannian interpretation of the Gauss–Newton algorithm. *To be presented at the 2nd IFAC Workshop on System Structure and Control, Prague*, 1992.

[21] M. Hazewinkel, Moduli and Canonical Forms for Linear Dynamical Systems II: The Topological Case, *Mathematical Systems Theory* 10, 363–385, 1977.

[22] M. Hazewinkel and R.E. Kalman, On invariants, canonical forms and moduli for linear constant finite dimensional dynamical systems, in : G. Marchesini and S.K. Mitter (eds), *Proceedings of the International Symposium on Mathematical System Theory, Udine, Italy*, Lecture Notes in Economics and Mathematical Systems 131, 48–60. Berlin: Springer Verlag, Berlin, 1976.

[23] R.E. Kalman, Algebraic geometric description of the class of linear systems of constant dimension, 8^{th} *Annual Princeton Conference on Information Sciences and Systems.* Princeton, N.J., 1974.

[24] R.E. Kalman, On partial realizations, transfer functions and canonical forms, *Acta Polytechnica Scandinavica*, Vol. Ma 31, 9–32, 1979.

[25] C.L. Lawson and R.J. Hanson, *Solving Least Squares Problems.* Englewood Cliffs: Prentice–Hall, 1974.

[26] K. Levenberg, A method for the solution of certain nonlinear problems in least squares, *Quart. Appl. Math.* 2, 164–168, 1944.

[27] A. Lichnewsky, Une méthode de gradient conjugué sur des variétés; application à certains problèmes de valeurs propres non linéaires, *Numer. Funct. Anal. and Optimiz.*, Vol. 1, 515–560, 1979.

[28] A. Lichnewsky, *Minimisation des Fonctionnelles Définies sur une Variété par la Méthode du Gradient Conjugué*, Thèse de Doctorat d'Etat. Paris: Université de Paris–Sud, 1979.

[29] L. Ljung, *System Identification: Theory for the User.* Englewood Cliffs: Prentice–Hall, 1987.

[30] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification.* Cambridge, Mass.: MIT Press, 1983.

[31] D.G. Luenberger, The Gradient Projection Method along Geodesics, *Management Science* 18, 620–631, 1972.

[32] D.W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *SIAM J. Appl. Math.* 11, 431–441, 1963.

[33] J.J. Moré, The Levenberg–Marquardt algorithm: implementation and theory, in: G.A. Watson (ed.) *Numerical Analysis, Lecture Notes in Mathematics* 630, 105–116. Berlin: Springer Verlag, 1977.

[34] A.J.M. van Overbeek and L. Ljung, On–line Structure Selection for Multivariable State Space Models, Report LiTH-ISY-I-0393. Linköping: Linköping University, 1980.

[35] A.J.M. van Overbeek and L. Ljung, On–line Structure Selection for Multi–Variable State–Space Models, *Automatica* 18, 529–543, 1982.

[36] R.L.M. Peeters, A Riemannian version of the Levenberg–Marquardt algorithm with application to system identification. Forthcoming.

[37] R.L.M. Peeters, *Ph.D. Thesis.* Forthcoming.

[38] G. Picci, Some Numerical Aspects of Multivariable Systems Identification, *Mathematical Programming Study* 18, 76–101, 1982.

[39] Yu.A. Rozanov, *Stationary Random Processes.* San Francisco: Holden–Day, 1967.

[40] H.L. Seal, The historical development of the Gauss linear model, *Biometrika* 54, 1–24, 1967.

[41] T. Söderström and P. Stoica, *System Identification.* New York: Prentice–Hall, 1989.

1991-1	N.M. van Dijk	On the Effect of Small Loss Probabilities in Input/Output Transmission Delay Systems
1991-2	N.M. van Dijk	Letters to the Editor: On a Simple Proof of Uniformization for Continious and Discrete-State Continious-Time Markov Chains
1991-3	N.M. van Dijk P.G. Taylor	An Error Bound for Approximating Discrete Time Servicing by a Processor Sharing Modification
1991-4	W. Henderson C.E.M. Pearce P.G. Taylor N.M. van Dijk	Insensitivity in Discrete Time Generalized Semi-Markov Processes
1991-5	N.M. van Dijk	On Error Bound Analysis for Transient Continuous-Time Markov Reward Structures
1991-6	N.M. van Dijk	On Uniformization for Nonhomogeneous Markov Chains
1991-7	N.M. van Dijk	Product Forms for Metropolitan Area Networks
1991-8	N.M. van Dijk	A Product Form Extension for Discrete-Time Communication Protocols
1991-9	N.M. van Dijk	A Note on Monotonicity in Multicasting
1991-10	N.M. van Dijk	An Exact Solution for a Finite Slotted Server Model
1991-11	N.M. van Dijk	On Product Form Approximations for Communication Networks with Losses: Error Bounds
1991-12	N.M. van Dijk	Simple Performability Bounds for Communication Networks
1991-13	N.M. van Dijk	Product Forms for Queueing Networks with Limited Clusters
1991-14	F.A.G. den Butter	Technische Ontwikkeling, Groei en Arbeidsproduktiviteit
1991-15	J.C.J.M. van den Bergh, P. Nijkamp	Operationalizing Sustainable Development: Dynamic Economic-Ecological Models
1991-16	J.C.J.M. van den Bergh	Sustainable Economic Development: An Overview
1991-17	J. Barendregt	Het mededingingsbeleid in Nederland: Konjunktuurgevoeligheid en effektiviteit
1991-18	B. Hanzon	On the Closure of Several Sets of ARMA and Linear State Space Models with a given Structure
1991-19	S. Eijffinger A. van Rixtel	The Japanese Financial System and Monetary Policy: a Descriptive Review
1991-20	L.J.G. van Wissen F. Bonnerman	A Dynamic Model of Simultaneous Migration and Labour Market Behaviour

1991-21	J.M. Sneek	On the Approximation of the Durbin-Watson Statistic in O(n) Operations
1991-22	J.M. Sneek	Approximating the Distribution of Sample Autocorrelations of Some Arima Processes in O(n) Operations
1991-23	B. Hanzon R. Hut	New Results on the Projection Filter
1991-24	R.J. Veldwijk E.R.K. Spoor M. Boogaard M.V. van Dijk	On Data Models as Meta Models, An Application Designers Point of View
1991-25	C. Camfferman	Some aspects of voluntary disclosure
1991-26	D.van der Wal	Monetary Policy Credibility: The Experience of the Netherlands
1991-27	J.A. Vijlbrief	Unemployment Insurance in a Disequilibrium Model for The Netherlands
1991-28	H.L.M. Kox	The "Non-Polluter gets paid" Principle for Third World Commodity Exports
1991-29	H. Tijms	A New Heuristic for the Overflow Probability in Finite-Buffer Queues
1991-30	B. Hanzon	On the Estimation of Stochastic Lineair Relations
1991-31	R.L.M. Peeters	Comments on Determining the Number of Zeros of a Complex Polynomial in a Half-Plane
1991-32	A.A.M. Boons H.J.E. Roberts F.A. Roozen	The Use of Activity-Based Costing Systems in a European Setting: a case study analysis
1991-33	J.C. van Ours	Union Growth in the Netherlands 1961-1989
1991-34	R. van Zijp	The Methodology of the Neo-Austrian Research Programme
1991-35	R.M. de Jong H.J. Bierens	On the Limit Behaviour of a Chi-Square Type Test if the Number of Conditional Moments Testes Approaches Infinity Preliminary Version
1991-36	K. Burger J.W. Gunning	Gender Issues in African Agriculture: Evidence from Kenya, Tanzania and Côte d'Ivoire
1991-37	M. Boogaard R.J. Veldwijk E.R.K. Spoor M.V. van Dijk	On Generalization in the Relational Model
1991-38	R. Dekker E. Smeitink	Preventive Maintenance at Opportunities of Restricted Duration